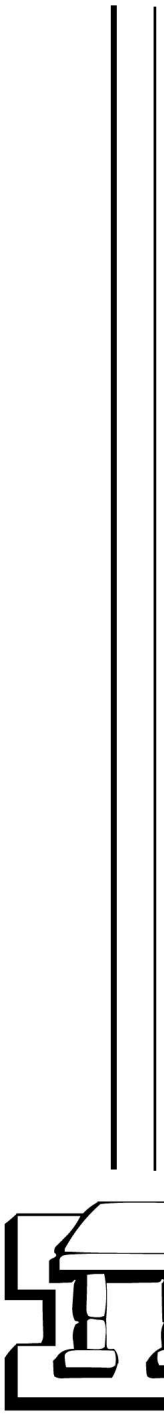
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO



FACULTAD DE ESTUDIOS SUPERIORES
IZTACALA

PROSA: UNA INTERFAZ DE WEB BASADA EN PERL PARA EL ANÁLISIS DE SECUENCIAS

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

B I Ó L O G O

PRESENTA:

JOSÉ MAURICIO MARIANO HERRERA CUADRA

DIRECTOR DE TESIS:

LIC. GUNNAR EYAL WOLF ISZAEVICH

LOS REYES IZTACALA, ESTADO DE MÉXICO. FEBRERO DE 2004

Agradecimientos

"Solamente los Espíritus del Aire saben qué me aguarda detrás de las montañas,
pero aún así sigo hacia delante…"

Antiguo proverbio esquimal

A **mi papá**, quien a pesar de mi rebeldía me ha apoyado en todo y puesto ejemplos: *"Levántate temprano…"*. A **mi mamá**, quien siempre ha estado ahí para sus hijos: *"¿Quieres que te haga taquito?"*. Al **Güero**, quien más me ha enseñado y echado la mano: *"¿Crees que las bacterias tengan sentimientos?"*. A **Juan Arturo**, quien siempre me ha soportado todo con prudencia: *"¿Cuándo me puedes pagar?"*. A **Gina**, por ser la hermana que nunca tuvimos en la familia: *"Yo no hablo si no está mi abogado…"*.

A **mi Tita**, porque siempre seré el más rubio y guapo para ti. A **mis abuelitos†**, porque educaron con amor y ejemplo. A **mis tías**, **tíos** y **primos** porque siempre me han echado porras. A **Poli**, porque eres la mejor prima del mundo, siempre estaremos juntos.

A los **Huampos**, por todos los logros compartidos y porque siempre me han apoyado en las buenas y en las malas, gracias de verdad. A **Gil**, por ser más que un hermano: *"¡Tú sabes qué pedo socio!…"*. A **Poncho**, por el empeño que pones en las cosas: *"Tómate una de… por… días"*. A **Jairo**, porque has dado todo sin esperar nada: *"¡Vamos a chupar!…"*. A **Serch**, porque estés donde estés te acuerdas de los amigos: *"Deberían de ver las mujeres que hay por acá…"*. A **Mau**, porque a pesar de tu acelere siempre estás contento: *"¡Leve, leve…!"*.

A **Beto**, porque hemos superado muchos miedos juntos, gracias por acompañarme en todas mis tarugadas: *"No hace falta que sea divertido para que sea divertido…"*. A **Cova**, porque me has dejado ser como un hermano mayor para ti: *"Tengo ganas de fisurear…"*. A **Karla**, por todo tu amor, paciencia y ternura, gracias por ser mi amiga y compañera en tantas travesuras: *"Ándale chuchis… ¡Forfafor!…"*. A **Marijó**, por tu inocencia: *"¿Y de qué platicas con esa vieja?"*. Al **Carnaza**, por tu nobleza: *"¿Me acompañas a dejar una rosa?"*. A **Pikos†**, porque sembraste un mensaje en nosotros y por tu filosofía: *"El mundo es de los audaces…"*. A **Fernanda**, por tu sencillez y ternura. A **Tebo**, **Laura**, **Martha** y la **Fam. Mar**, porque siempre nos recibieron con los brazos abiertos aunque solo fuera para echar desmadre. A **Montse**, por todos los momentos difíciles, porque encontré lo mejor de mí gracias a ti, siempre serás *"la poeta ahogada en sentimiento"*. A **Juliana**, por la magia de tantos momentos, por tu amistad y sinceridad: *"No existe la mujer ideal…"*. A **Jada**, por tu amistad a pesar del tiempo y la distancia.

A **Armando**, por todo lo que aprendí desde niño: *"¡Tensa, se atoró el stopper!"*. A **Omar**, por seguir siendo impredecible: *"Mi querido salta pa' tras…"*. A **Lorena**, por querer tanto a Omar. A **Hans**, por tu actitud positiva ante la vida, gracias por no dejar que nada te detenga: *"¿Qué onda dude? ¿Run, run?"*. A **los Pérez** (**Rubén**, **Daniel**, **José** y **Chucho**), por tantas pláticas, aventuras y convivencias. A **Gorka**, por tu hermosa familia y por cuidarme como un hermano mayor: *"¿Cómo está el bestia de tu hermano?"*. A **Vicente**, por tu generosidad y porque siempre has estado ahí.

Al **Grupo de Misiones "Nochi Ica Toteco"**, por las vivencias compartidas, las responsabilidades, los enojos, y por formarme un poco de criterio y carácter. A **Bosco**, por la enseñanza, confianza y apoyo, pero sobretodo por ser mi amigo y hermano: *"Sí, es cabroncito…"*. A la **Trucha**, por poner el ejemplo y la confianza: *"¡Ándele mi Yogui!..."*. A **Ana Laura** y **Fernando**, por su gran amistad y lealtad. A **Vero** y **Máfer**, por la amistad dentro y fuera de misiones.

A **Ginna**, por ser mi inseparable amiga, por tu lealtad y preocupación, por escucharme en todo momento y por el empeño que pones en todo lo que te propones. A **Octavio**, por nuestra gran amistad, tu dedicación y tu nobleza, por todos los momentos de hilaridad (la mayoría del tiempo…), y porque eres el único que conozco que puede soportar el mismo disco de Slayer… ¡por más de 6 horas!: *"¡Up the irons!"*. A **Mariana**, por todo tu cariño, creatividad y explosividad. A **Monse**, por tu amistad. Al **Dr. Genoma (Octavio G.)**, por todos los buenos momentos, porque más vale tarde que nunca, gracias por la amistad mas allá del orgullo: *"¿Qué pues mi rey?"*. A **Sergio**, porque siempre has sacrificado todo por tus amigos, gracias por tu visión de la vida, por el tiempo, la paciencia, la poesía y la confianza: *"¡No mames güey!… ¿Tu eres chingón no?…"*. A **Rodolfo**, por tu amistad sincera, la confianza y todos los buenos momentos: *"¡Se cae el abuelo!"*. A **Salvador**, por ser mi padre académico, por enseñarme a plantear bien las preguntas, por todo tu conocimiento y por compartirlo, por las excelentes mesas redondas, por el sarcasmo y sobretodo por tu amistad y comprensión. A **Nacho**, porque predicas con el ejemplo, gracias por la confianza y el apoyo para alcanzar mis metas. A **Gunnar**, por darme el último empujón para estudiar bioinformática y que todo esto sucediera, gracias de verdad, eres muy buen amigo y maestro: *"¡Pero yo no se nada de biología!…"*. A **Cházaro**, por tu amistad y las oportunidades. A **Roberto Rico**, por todo su conocimiento y su amistad. A **Eugenio**, por la confianza y las porras: *"¡Pinche Mau, erraste la carrera Mau!"*. A **Carmelita** y **Celia**, porque siempre están al pie del cañón, son las mejores y nuestras consentidas. A **Miguel**, **Tintín** y **Rigo**, por esa noche de fiesta en Tehuacán (¡qué manera de beber!): *"¡Competencia de lagartijas!… y después… ¡todos al balcón!"*. A **Chibebo**, por la maraña de genialidades que tienes en el cerebro: *"¿Esternocleidomastoideo?"*. A **Adriana**, por desconcentrarme de la computadora cada que podías: *"¿Qué onda Mau, mucha chamba?"*. A **Antalia**, por el cariño, la amistad y las porras a pesar de la distancia. A **Luis Oliver**, **Noé**, **Xóchitl**, **Ricardo** y **Víctor**, por los buenos momentos durante los congresos. A **Marcos**, por todo lo que has cambiado. A **Fido**, por el espíritu escalador. Al **Borrego**, por tu nobleza. A **Verónica**, por todo el boxeo que pudimos tener: *"¿Qué traes maldita güera? ¡Pelea!"*. A **Paty** y **Cecilia**, por su amistad. A **Andrea**, por abrirme los ojos. A **Manuel**, **Erika**, **Checo**, **Israel**, **Alina**, **Ale**, **Paola**, **David**, **Compadre**, **Ivonne Herrera**, **Martha**, **Güera**, **Ivonne Miranda**, **César**, **Bárbara**, **Topeka**, **Enrique**, **Toño**, **Grillo**, **Jován**, **Hugo**, **Betzabet**, por todas las clases y laboratorios a los que asistimos (y a los que no…). Y a toda la gente que he conocido a través de los años en la **FES Iztacala** (**Susana Robles**, **Gloria Luz Paniagua**, **Erasmo Negrete**, **Diego Arenas**, **Paty Dávila**, **Marcela Ibarra**, **Alex Juárez**, **Rafael Lira**, **Rocío Mayorga**, **Pablo García**, **Laura Castañeda**, **Margarita Canales**, **Jonás Barrera**, **Rodolfo de la Torre**, **Claudia Diez**, **Ángel Durán**, **Diódoro Granados**, **Luis Antonio**, **Peter Mueller**, **Elías Piedra**, **Arnulfo Reyes**, **Vicente Sandoval**, etc.), por habernos topado aunque fuera en el pasillo y aun así me han dado ánimos y sonrisas.

A mis compañeros de **la Congeladora del Área 2 de la UFRAM** (**Jorge**, **Elisa**, **Güicho**, **Jonathan**, **Apolinar**, **Christian**), por todos los momentos divertidos e interesantes en el salón. A mis demás compañeros de la **UFRAM** (**Alicia**, **Diego**, **Ursula**, **Esther**, **Fabián**, **Yeyo**, **Sandra**, etc.), por su amistad sincera. A mis maestros de la **UFRAM** (**Ocaña**, **Sergio**, **Lorena**, **Claudia**), porque fomentaron en mí una mayor pasión hacia la biología.

A mis compañeros del **Colegio Cristóbal Colón** (**Mike**, **Boogie**, **Pavel**, **Andrik**, **Nacho**, etc.), con quienes disfrute de todo lo que es posible durante la secundaria. A **Miguel Pavón**, por tu amistad y todas las ilusiones que no cumplimos: *"¿Qué onda gordo, vamos al Everest?"*. A **Elvira**, por todos los pleitos que hemos tenido con mucho cariño. A **Nelly**, porque siempre te acuerdas de mi cumpleaños aunque yo no me acuerde del tuyo: *"¿Es un día antes?"*. A **Maribé**, por todo el tango que bailamos sobre las mesas. Al **Ganso** y **Fam. González**, por recibirme como uno más en su casa. A **Carol**, **Dana** y **Fam. López Breña**, por adoptarme como a un hermano. A **Jordy**, por inventar cuanta estupidez era posible en clase: *"¿Señora con C?"*. A **Renán**, porque supiste ser amigo. A **Hans Román**, por enseñarme a tocar la guitarra de forma un poco más decente: *"Primero vamos a ejercitar los dedos por 45 minutos… Sí, pero… ¿y la canción que quiero tocar?"*. A **Manuel Mijares**, por darme ánimos cuando pensaba que ya no podía más. A **Manuel Cortazar†**, por su amistad y vocación.

A **Aldo** y **Hubert**, con quienes tuve la infancia más entretenida y divertida posible, casi nos cambiaban los pañales juntos. A mis compañeros del **Colegio Fátima** (**Jonathan**, **Marifer C**., **Ale**, **Andrés**, **Karen**, **Iván**, **Juan Carlos**, **Irma**, **Marifer R**., **Chuchú**, **Piero**, **Lorena**, **Imelda**, **Sergio**, **Oscar**, **Champi**, etc.), por tantos años que crecimos juntos. A **Marthita**, porque fuiste el primer amor. A la **Miss Adriana** por tantos años de cariño y dedicación hacia nosotros.

A la **banda escaladora** (**Paco**, **César**, **Elsa**, **Carlos Carsolio**, **Jorge Colín**, **Alan**, **Aarón**, **Julieta**, **Cristel**, **Viviana**, **Manuel**, **Carla**, **Higinio**, **Pablo**, **Héctor**, **Andrés**, **Bonfilio**, **Charly**, **Oscar**, **Farfán**, **Raúl**, **Jorge Wingartz**, **Camacho**, **Fernando**, **Rodulfo**, **Carlos Suárez**, **Christian**, **Jean-Louis**, **Clarita**, **Periquito**, etc.), con quienes he compartido muchos momentos en situaciones diversas.

A los ingenieros de **Yahoo! México** (**Daniel**, **Moad**, **Memo**, **Rubén**, **Lucas** y **Porras**), por su amistad, apoyo, confianza y oportunidades. A los demás Yahoo!'s (**Mike**, **Tania**, **Alma**, **Mauricio**, **Jorge**, **Vanesa**, **Lorena**, **Yamile**, **José Manuel**, **Julieta**), por su amistad, su compañerismo y por tener la camiseta bien puesta: *"Do you Yahoo!?"*.

A la **Fam. Pérez-Vertti**, por el cariño de tantos años. A la **Fam. Dattoli**, por su amistad. A **mi tía Lola**, porque simplemente eres pocamadre! A la **Fam. Abbadié**, por permitirme ser uno de ellos.

A **Kurt Diemberger**, por toda la poesía e inspiración que emanan sus libros. A **Mark Twight**, por su disciplina y manera de percibir el alpinismo moderno.

Al finalizar esta "regresión" han aparecido múltiples sentimientos en mi mente y corazón. Me considero afortunado de haber podido coincidir con tanta gente en el complicado laberinto de la vida, les agradezco a todos los que aquí he mencionado y a los que no también. Gracias a ustedes soy quien soy.

Finalmente, quiero agradecer de manera especial a las montañas, porque siempre han estado y estarán ahí. Gracias por permitirme regresar. La energía que les he ofrecido me ha sido devuelta multiplicada por diez.

El secreto está en la música…

Índice

1. Introducción

1.1 El análisis de secuencias y su importancia

La segunda mitad del siglo XX fue testigo de increíbles avances en biología molecular y tecnología computacional. Tan solo 50 años después de haberse identificado la estructura química del DNA (1953), la secuencia del genoma humano ha sido determinada y puede ser descargada a una computadora lo suficientemente pequeña como para caber en nuestra mano. El ritmo de la ciencia puede ser verdaderamente vertiginoso. ¿Qué podemos hacer cuando literalmente tenemos el libro de la vida en la palma de nuestra mano? Por supuesto que lo leemos. Desafortunadamente, es mucho más fácil leer el libro de la vida que entenderlo, por lo que uno de los grandes retos del siglo XXI será desenmarañar sus misterios. Un acercamiento particularmente fructífero hacia descifrar el libro de la vida ha sido a través de estudios comparativos, como aquellos entre el ratón y el humano. Las comparaciones entre los genomas del humano y el ratón muestran lo poco que han cambiado desde que los humanos y los ratones compartieron por última vez un ancestro común hace 75 millones de años aproximadamente (Bedell *et al*., 2003).

La información funcional y hereditaria de un organismo se encuentra almacenada en moléculas de DNA, RNA y proteínas, todas estas macromoléculas son cadenas lineales compuestas de moléculas más pequeñas. Estas macromoléculas son ensambladas a partir de un alfabeto fijo de compuestos químicos bien conocidos: el DNA está formado por cuatro desoxirribonucleótidos, el RNA está formado por cuatro ribonucleótidos y las proteínas están formadas por 20 aminoácidos. Debido a que estas macromoléculas son cadenas lineales de compuestos definidos, pueden ser representadas como secuencias de símbolos. Estas secuencias pueden ser entonces comparadas para encontrar similitudes que sugieran que las moléculas están relacionadas por su forma o función (Gibas *et al*., 2001).

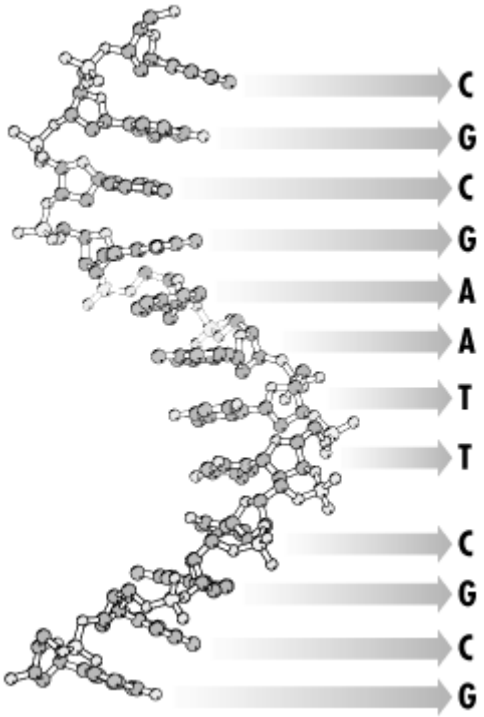


Figura 1. Representación del DNA en una secuencia de símbolos (Gibas *et al*., 2001)

Es importante recordar que una secuencia biológica (DNA, RNA o proteína) posee una función química, pero cuando esta es reducida a un código de letras sencillas funciona también como una etiqueta única, casi como un código de barras. Desde el punto de vista de la tecnología de la información, la información de las secuencias es invaluable. La etiqueta de la secuencia puede ser aplicada a un gen, su producto, su función, su rol en el metabolismo celular, etc. (Gibas *et al*., 2001).

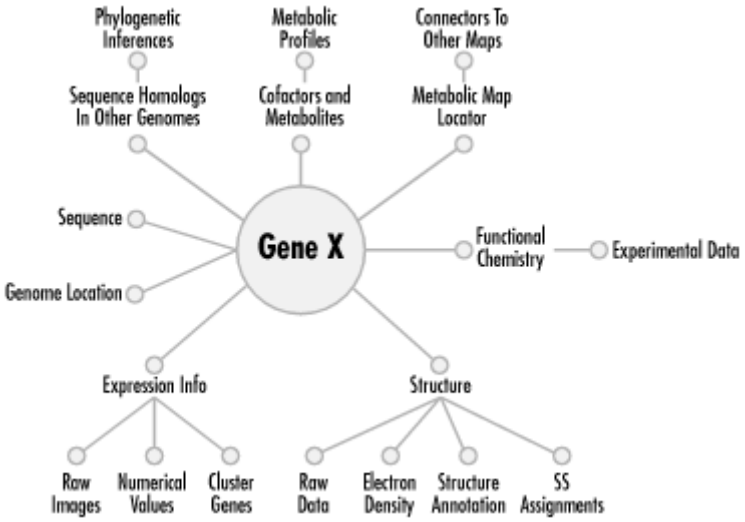


Figura 2. Posible información asociada a un solo gen (Gibas *et al*., 2001)

Sin embargo, la cuestión más importante acerca de estas etiquetas, es que no solamente identifican un gen particular; también contienen patrones biológicamente significativos, que permiten comparar diferentes etiquetas, conectar información, y hacer inferencias. Así que no solamente las etiquetas pueden conectar toda la información acerca de un gen, éstas pueden servir para conectar información sobre genes que son ligera o drásticamente diferentes en su secuencia (Gibas *et al*., 2001).

Los datos de las secuencias de genes son el más abundante tipo de información, y existe un gran conjunto de métodos y herramientas computacionales que pueden ayudar a analizar los patrones contenidos en dicha información. No es mera coincidencia que las secuencias de genes de plantas, animales y microorganismos muestren complejos patrones de similitud entre ellas. Este es uno de los aspectos más fascinantes del estudio de la evolución. De hecho, algunos biólogos moleculares están convencidos de que el entendimiento de la evolución de las secuencias es el primer paso hacia el entendimiento mismo de la evolución. La comparación de secuencias de genes, o análisis de secuencias biológicas, es uno de los procesos utilizados para comprender la evolución de las secuencias. Es una disciplina importante dentro de la biología computacional y la bioinformática (Leon *et al*., 2003).

1.2 La investigación biológica a través de la World Wide Web

La Internet ha cambiado completamente la forma en que los científicos buscan e intercambian información. La información que antes era comunicada en papel ahora es digitalizada y distribuida a partir de bases de datos centralizadas, las revistas ahora son publicadas "en línea", y casi cualquier grupo de investigación posee un Sitio Web que ofrece de todo, desde publicaciones hasta descargas de software y servicios automatizados de procesamiento de datos (Gibas *et al*., 2001).

Existen docenas de bases de datos en Internet, y muchas interfaces de Web que proveen acceso a este conjunto de datos. Sin embargo, pocas de estas interfaces llevan a cabo un análisis profundo de la

información, por lo que los usuarios se enfrentan frecuentemente con la necesidad de utilizar múltiples y diferentes interfaces, para obtener el tipo de información que necesitan (Stein, 2002).

Los científicos utilizan los servicios Web en Internet para la mayoría de los análisis de datos hoy en día. Esto es debido a su accesibilidad, interfaz simple de documentos y formularios, y frecuentemente servicios gratuitos que proveen muchas herramientas de análisis y bases de datos actualizadas. Aún cuando las interfaces de Web para los análisis de biología molecular no siempre son la mejor opción, si éstas son capaces de realizar el trabajo, son preferibles a un programa ejecutándose bajo algún sistema operativo específico (Gilbert, 2002).

La mayoría de los usuarios encuentran problemas al utilizar programas para el análisis de secuencias. No solamente son difíciles de aprender debido a los parámetros, sintaxis y semántica, sino a que muchos son diferentes. Debido a esto, los programadores se han dedicado a construir interfaces de Web que simplifiquen el aprendizaje y utilización de dichos programas, un claro ejemplo de dicha tendencia son interfaces como: *Virtual PCR* (Lexa *et al*., 2001) y *WebPHYLIP* (Lim *et al*., 1999). Inclusive se han desarrollado sistemas avanzados, tales como: *Pise*, que permiten la generación de interfaces de Web a partir de programas de biología molecular más sencillos (Letondal, 2001; Gilbert, 2002).

1.3 La base de datos PROSITE

Un motivo es una región o porción de una secuencia de proteína que posee una estructura específica y es funcionalmente significativa. Las familias de proteínas a menudo son caracterizadas mediante uno o más de tales motivos. La detección de motivos en proteínas es un problema importante puesto que los motivos portan y regulan varias funciones, y la presencia de motivos específicos puede ayudar a clasificar una proteína (Narasimhan *et al*., 2002).

PROSITE es una colección de descriptores de motivos dedicada a la identificación de familias de proteínas y dominios. Los descriptores de motivos utilizados en PROSITE son patrones o perfiles, los cuales han sido derivados a partir de alineamientos múltiples de secuencias homólogas. Esto proporciona a estos descriptores de motivos la notable ventaja de identificar relaciones distantes entre secuencias que hubieran pasado inadvertidas mediante alineamiento simple de secuencias. Los patrones y perfiles poseen tanto ventajas como desventajas, los cuales definen su área de aplicación (Sigrist *et al*., 2002).

Los patrones de PROSITE son construidos a partir de alineamientos de secuencias relacionadas, las cuales son obtenidas de una variedad de fuentes: de una familia de proteínas bien caracterizada; de la literatura; de resultados de la búsqueda de secuencias en SWISS-PROT y TrEMBL; o de agrupamiento de secuencias. Dichos alineamientos son revisados en búsqueda de regiones conservadas, particularmente para las familias de proteínas caracterizadas, cuya actividad catalítica o su anclaje a un sustrato puede haber sido demostrado experimentalmente. Se crea un patrón central en forma de una expresión regular[1] que especifica cuál o cuáles aminoácidos pudieran estar presentes en cada posición. Una vez que el patrón central está hecho, es probado contra secuencias en SWISS-PROT. Si la serie correcta de proteínas concuerda con este patrón, entonces es almacenado; si falla en recoger algunos miembros de la familia o recoge demasiadas proteínas no relacionadas, el patrón es refinado y vuelto a probar hasta ser óptimo (Mulder *et al*., 2001).

Los patrones poseen muchas ventajas, pero también tienen sus limitaciones a través de secuencias completas, por lo que PROSITE también crea perfiles, para complementar a los patrones. Para estos, el proceso también comienza con alineamientos múltiples de secuencias; y después utiliza una tabla de comparación de símbolos para convertir distribuciones libres de residuos en pesos, resultando en una

[1] Las expresiones regulares son cadenas de texto que describen patrones, utilizados para representar una serie de cadenas. Estas son mucho más elaboradas y poderosas que los comodines, pero también son mucho más complejas.

tabla de pesos de posición especifica. Una tabla de comparación de símbolos comprende valores que describen la comparación entre pares de aminoácidos. La tabla tiene un valor para medir la calidad de la concordancia de cada posible par de aminoácidos, y se utiliza para proveer puntajes sobre la probabilidad de que un aminoácido sea reemplazado por otro en una posición particular dentro del alineamiento de la secuencia. Estos números son utilizados para calcular un puntaje de similitud del alineamiento entre el perfil y las secuencias en SWISS-PROT; un alineamiento con un puntaje de similitud igual o mayor constituye un acierto. El perfil es entonces refinado hasta que solamente la serie deseada de secuencias de proteína da un puntaje por encima del umbral para el perfil (Mulder *et al*., 2001).

PROSITE es un método para determinar cual es la función de proteínas no caracterizadas que han sido traducidas de secuencias de cDNA o DNA genómico. Esta base de datos está elaborada de tal forma que con herramientas computacionales apropiadas, pueda ser rápido y factible el identificar a qué familia conocida de proteínas (si la hay) pertenece una nueva secuencia. En algunos casos, la secuencia de una proteína desconocida se encuentra lejanamente relacionada con cualquier proteína de estructura conocida para poder detectar su semejanza por medio de alineamiento de secuencias completas. Sin embargo, puede ser identificada por la presencia en su secuencia de un bloque particular de tipos de residuos, diversamente conocidos como patrones, motivos, firmas, o huellas digitales. Estos motivos sobresalen debido a los requerimientos particulares en la estructura de regiones específicas de una proteína, los cuales pudieran ser importantes, por ejemplo, por sus propiedades de anclaje, o por su actividad enzimática (Tisdall, 2001).

1.3.1 Descriptores de motivos de PROSITE

A continuación se muestran ejemplos de los 2 tipos de descriptores de motivos contenidos en la base de datos PROSITE. Su explicación es detallada más adelante.

1.3.1.1 Descriptor de tipo “patrón”

```
ID PPASE; PATTERN.
AC PS00387;
DT NOV-1990 (CREATED); NOV-1995 (DATA UPDATE); NOV-1995 (INFO UPDATE).
DE Inorganic pyrophosphatase signature.
PA D-[SGN]-D-[PE]-[LIVM]-D-[LIVMGC].
NR /RELEASE=32,49340;
NR /TOTAL=16(16); /POSITIVE=11(11); /UNKNOWN=0(0); /FALSE_POS=5(5);
NR /FALSE_NEG=0; /PARTIAL=2;
CC /TAXO-RANGE=A?EP?; /MAX-REPEAT=1;
CC /SITE=1,magnesium; /SITE=3,magnesium; /SITE=6,magnesium;
DR P21216, IPYR_ARATH, T; P37980, IPYR_BOVIN, T; P17288, IPYR_ECOLI, T;
DR P44529, IPYR_HAEIN, T; P13998, IPYR_KLULA, T; P19117, IPYR_SCHPO, T;
DR P37981, IPYR_THEAC, T; P19514, IPYR_THEP3, T; P38576, IPYR_THETH, T;
DR P00817, IPYR_YEAST, T; P28239, IPY2_YEAST, T;
DR P19371, IPYR_DESVH, P; P21616, IPYR_PHAAU, P;
DR P09167, AERA_AERHY, F; P12351, CYP1_YEAST, F; P24653, Y101_NPVOP, F;
DR P37904, YCEI_ECOLI, F; P39303, YJFU_ECOLI, F;
3D 1PYP;
DO PDOC00325;
//
```

1.3.1.2 Descriptor de tipo “perfil” (matriz)

```
ID GLOBIN; MATRIX.
AC PS01033;
DT JUN-1994 (CREATED); DEC-2001 (DATA UPDATE); DEC-2001 (INFO UPDATE).
DE Globins profile.
MA /GENERAL_SPEC: ALPHABET='ABCDEFGHIKLMNPQRSTVWYZ'; LENGTH=154;
MA /DISJOINT: DEFINITION=PROTECT; N1=1; N2=154;
```

```
MA /NORMALIZATION: MODE=1; FUNCTION=LINEAR; R1=-0.8705306; R2=0.0209303; TEXT='-LogE';
MA /CUT_OFF: LEVEL=0; SCORE=424; N_SCORE=8.0; MODE=1; TEXT='!';
MA /CUT_OFF: LEVEL=-1; SCORE=353; N_SCORE=6.5; MODE=1; TEXT='?';
MA /DEFAULT: D=-20; I=-20; MI=-210; MD=-210; IM=0; DM=0;
MA /I: I=-6;
MA /M: SY='A'; M=7,-7,-8,-10,-10,-8,3,-12,-4,-8,-6,-4,-6,-10,-10,-10,3,4,3,
-14,-10,-10; D=-6;
MA /I: I=-6; MI=-59; MD=-59;
MA /M: SY='H'; M=1,-3,-21,0,-6,-20,0,2,-16,-10,-16,-10,-4,0,-8,-12,-2,-9,-11,
-23,-13,-8; D=-6;

<eliminado por brevedad>

MA /M: SY='H'; M=-1,4,-18,5,3,-19,-10,9,-20,8,-16,-9,3,-10,2,8,-2,-7,-14,-19,-6,1; D=-5;
MA /I: I=0; MI=*;
NR /RELEASE=40.7,103373;
NR /TOTAL=797(796); /POSITIVE=796(795); /UNKNOWN=0(0); /FALSE_POS=1(1);
NR /FALSE_NEG=0; /PARTIAL=3;
CC /MATRIX_TYPE=protein_domain;
CC /SCALING_DB=reversed;
CC /AUTHOR=P_Bucher;
CC /TAXO-RANGE=??EP?; /MAX-REPEAT=9;
CC /FT_KEY=DOMAIN; /FT_DESC=GLOBIN;
DR P04252, BAHG_VITST, T; Q03331, FHP_CANNO , T; P39676, FHP_YEAST , T;
DR P02212, GLB1_ANABR, T; P19363, GLB1_ARTSX, T; P14805, GLB1_CALSO, T;
DR P02221, GLB1_CHITH, T; P02216, GLB1_GLYDI, T; P20412, GLB1_LAMSP, T;
DR P41260, GLB1_LUCPE, T; P08924, GLB1_LUMTE, T; P21197, GLB1_MORMR, T;

<eliminado por brevedad>

DR P42430, YKYB_BACSU, F;
3D 1VHB; 2VHB; 3VHB; 1HBG; 2HBG; 1BOB; 1EBT; 1FLP; 1MOH; 1HBI; 2HBI; 3HBI;
3D 3SDH; 4HBI; 4SDH; 5HBI; 6HBI; 7HBI; 1ECA; 1ECD; 1ECN; 1ECO; 1VRE; 1VRF;
3D 2LHB; 3LHB; 1DM1; 1MBA; 2FAL; 2FAM; 3MBA; 4MBA; 5MBA; 1SCT; 1HLB; 1HLM;
3D 1OUT; 1OUU; 1A4F; 1FSX; 1HDA; 1CG5; 1CG8; 1IBE; 2DHB; 2MHB; 1AOO; 1AO1;
3D 1AOU; 1AOV; 1AOW; 1AOX; 1AOY; 1AOZ; 1A3N; 1A3O; 1A9W; 1ABW; 1ABY; 1AJ9;
3D 1AXF; 1B86; 1BAB; 1BBB; 1BIJ; 1BUW; 1BZO; 1BZ1; 1BZZ; 1CLS; 1CMY; 1COH;
3D 1DSH; 1DXT; 1DXU; 1DXV; 1FDH; 1GBU; 1GBV; 1GLI; 1HAB; 1HAC; 1HBA; 1HBB;
3D 1HBS; 1HCO; 1HDB; 1HGA; 1HGB; 1HGC; 1HHO; 1NIH; 1QI8; 1QSH; 1QSI; 1RVW;
3D 1SDK; 1SDL; 1THB; 1VWT; 2HBC; 2HBD; 2HBE; 2HBF; 2HBS; 2HCO; 2HHB; 2HHD;
3D 2HHE; 3HHB; 4HHB; 6HBW; 1SPG; 1HDS; 1HBH; 1PBX; 1QPW; 2PGH; 1HBR; 1CBL;
3D 1CBM; 1ITH; 1D8U; 1CQX; 1GDI; 1GDJ; 1GDK; 1GDL; 1LH1; 1LH2; 1LH3; 1LH5;
3D 1LH6; 1LH7; 2GDM; 2LH1; 2LH2; 2LH3; 2LH5; 2LH6; 2LH7; 1BIN; 1FSL; 1LHS;
3D 1LHT; 1EMY; 1MBS; 1AZI; 1BJE; 1DWR; 1DWS; 1DWT; 1HRM; 1HSY; 1RSE; 1WLA;
3D 1XCH; 1YMA; 1YMB; 1YMC; 2MM1; 1O1M; 1O2M; 1O3M; 1O4M; 1O5M; 1O6M; 1O7M;
3D 1O8M; 1O9M; 11OM; 111M; 112M; 1A6G; 1A6K; 1A6M; 1A6N; 1ABS; 1AJG; 1AJH;
3D 1BVC; 1BVD; 1BZ6; 1BZP; 1BZR; 1CH1; 1CH2; 1CH3; 1CH5; 1CH7; 1CH9; 1CIK;
3D 1CIO; 1CO8; 1CO9; 1CPO; 1CP5; 1CPW; 1CQ2; 1DO1; 1DO3; 1DO4; 1DO7; 1DTI;
3D 1DTM; 1DUK; 1DUO; 1DXC; 1DXD; 1EBC; 1F63; 1F65; 1F6H; 1FCS; 1HJT; 1IOP;
3D 1IRC; 1JDO; 1LTW; 1MBC; 1MBD; 1MBI; 1MBN; 1MBO; 1MCY; 1MGN; 1MLF; 1MLG;
3D 1MLH; 1MLJ; 1MLK; 1MLL; 1MLM; 1MLN; 1MLO; 1MLQ; 1MLR; 1MLS; 1MLU; 1MOA;
3D 1MOB; 1MOC; 1MOD; 1MTI; 1MTJ; 1MTK; 1MYF; 1MYM; 1OBM; 1OFJ; 1OFK; 1SPE;
3D 1SWM; 1TES; 1VXA; 1VXB; 1VXC; 1VXD; 1VXE; 1VXF; 1VXG; 1VXH; 1YOG; 1YOH;
3D 1YOI; 2CMM; 2MB5; 2MBW; 2MGA; 2MGB; 2MGC; 2MGD; 2MGE; 2MGF; 2MGG; 2MGH;
3D 2MGI; 2MGJ; 2MGK; 2MGL; 2MGM; 2MYA; 2MYB; 2MYC; 2MYD; 2MYE; 2SPL; 2SPM;
3D 2SPN; 2SPO; 4MBN; 5MBN; 1M6C; 1M6M; 1MDN; 1MNH; 1MNI; 1MNJ; 1MNK; 1MNO;
3D 1MWC; 1MWD; 1MYG; 1MYH; 1MYI; 1MYJ; 1PMB; 1YCA; 1YCB; 1MYT; 1ASH;
DO PDOC00793;
//
```

1.3.2 Definición de los campos de PROSITE

Los códigos de campo de la base de datos PROSITE ayudan a organizar su información para legibilidad humana y procesamiento computacional. Existen varios códigos de campo dentro de un solo registro; los cuales son representados con una abreviación de dos letras. La siguiente tabla muestra las definiciones y descripciones para estos códigos (Leon *et al.*, 2003).

Tabla 1. Definición de los campos de PROSITE

Campo	Definición	Descripción
ID	Identificación	El segundo elemento indica el tipo de registro: • PATTERN • MATRIX • RULE
AC	Numero de acceso	PSnnnnn.
DT	Fecha	Fecha de creación o ultima modificación del registro.
DE	Descripción corta	Información descriptiva sobre el contenido del registro.
PA	Patrón	La definición de un patrón de PROSITE.
MA	Matriz/perfil	La definición de un perfil/matriz de PROSITE.
RU	Regla	La definición de una regla de PROSITE.
NR	Resultados numéricos	Contiene información relevante a los resultados de la búsqueda con un patrón en la base SWISS-PROT completa. Se utilizan los siguientes calificadores: • /RELEASE • /TOTAL • /POSITIVE • /UNKNOWN • /FALSE_POS • /FALSE_NEG • /PARTIAL
CC	Comentarios	Varios tipos de comentarios. Se utilizan los siguientes calificadores: • /TAXO-RANGE • /MAX-REPEAT • /SITE • /SKIP-FLAG • /MATRIX_TYPE • /SCALING_DB • /AUTHOR • /FT_KEY • /FT_DESC
DR	Referencias cruzadas hacia SWISS-PROT	Se utilizan como apuntadores hacia registros de SWISS-PROT.
3D	Referencias cruzadas hacia PDB	Se utilizan para enlistar los registros del Protein Data Bank.
DO	Apuntador hacia el archivo de documentación	Contiene un apuntador hacia la documentación de PROSITE que describe este registro.
//	Línea de terminación	Designa el final de un registro.

1.3.3 Disponibilidad de PROSITE

PROSITE se encuentra disponible como una serie de archivos de texto que proveen los datos, además de documentación. El sitio de PROSITE (*http://www.expasy.org/prosite/*) está provisto de una interfaz de usuario que permite indagar en la base de datos y examinar la documentación. La base de datos también puede obtenerse para instalación local a través del sitio FTP de PROSITE. Su utilización es gratuita para usuarios no comerciales (Sigrist *et al*., 2002; Tisdall, 2001).

1.3.4 Aplicaciones que utilizan PROSITE

Existen algunos programas que ayudan a la búsqueda de patrones y perfiles de PROSITE en secuencias de proteína. Su descripción se encuentra en la siguiente tabla (Sigrist *et al*., 2002):

Tabla 2. Aplicaciones que utilizan PROSITE

Aplicación	Descripción
ps_scan	Busca uno o varios motivos de PROSITE dentro de una o varias secuencias de proteína. Disponible en *ftp://ftp.expasy.org/databases/prosite/tools/*
PFTOOLS	Consta de varios programas que sirven para construir perfiles y/o buscar perfiles o bibliotecas de perfiles dentro de una secuencia o una biblioteca de secuencias. Disponible en *ftp://ftp.expasy.org/databases/prosite/tools/*
ScanProsite	Permite buscar dentro de una secuencia de proteína la presencia de motivos de PROSITE o buscar en las bases de datos SWISS-PROT, TrEMBL y/o PDB por la presencia de un patrón que provenga de PROSITE o del usuario. También permite visualizar la posición de un motivo de PROSITE o del usuario en la estructura 3D (si se conoce) de la proteína concordante. Recientemente se añadió la posibilidad de evaluar la especificidad de un patrón utilizándolo para buscar en una versión aleatorizada de la base de datos SWISS-PROT. Se encuentra disponible en *http://www.expasy.org/tools/scanprosite*
ProfileScan	Busca perfiles de PROSITE dentro de una secuencia de proteína. Disponible en *http://hits.isb-sib.ch/cgi-bin/PFSCAN*

1.4 Perl y su aplicación en Bioinformática

Una gran parte de la Biología Computacional consiste de tareas frecuentes de procesamiento de textos, tales como la manipulación de cadenas, concordancia de expresiones regulares, traducción de archivos, e interconversión de formato de datos. Por consiguiente, muchos desarrolladores en la comunidad bioinformática hacen uso extenso del lenguaje de programación Perl, el cual sobresale en dichas tareas (Chervitz *et al*., 1998).

Perl es popular entre los biólogos debido a su carácter práctico. La información biológica en las computadoras tiende a estar organizada en archivos de texto o en bases de datos relacionales[2]. Cualquiera de estas fuentes de datos es fácil de manejar con programas en Perl. Perl se ha convertido en una especie de fenómeno en el área, puesto que muchos biólogos lo encuentran como un lenguaje fácil de aprender que posee muchas de las herramientas que ellos necesitan: en particular su soporte para el procesamiento de textos y expresiones regulares lo hacen adecuado para tareas complejas de traducción de textos (comunes en bioinformática) (Chervitz *et al*., 1998).

Perl ha demostrado ser un poderoso y fácil lenguaje de alto nivel para programación, desarrollo orientado a objetos, y desarrollo rápido de prototipos para software bioinformático. Los programas en Perl pueden ser vistos como modelos para bio-objetos y conceptos que puedan ser reimplementados en otros lenguajes de programación (Chervitz *et al*., 1998).

Perl ha madurado de un simple lenguaje de "script" a un poderoso ambiente de programación tanto para el estilo procedimental como para el orientado a objetos. Mientras que sigue siendo utilizado para crear programas simples "desechables", también se utiliza para diseñar aplicaciones complejas, modulares, bien documentadas y mantenibles. La facilidad de utilización de Perl para una variedad de tareas, tanto de alto nivel como para programación de CGI[3], es inigualable (Chervitz *et al*., 1998).

[2] Una base de datos es una colección organizada de datos. Una base de datos relacional organiza los datos en tablas.
[3] Common Gateway Interface, es una tecnología que permite a los servidores de Web proveer aplicaciones en línea.

Esto sucede debido a que Perl es mucho más poderoso de lo que la gente piensa, pero su poder proviene de una manera interesante. Perl posee el inusual don de unir cosas. Esto lo hace mejor que cualquier otro lenguaje, y lo hace en muchos niveles diferentes. He aquí algunos ejemplos:

- Perl puede unir sitios Web y bases de datos a través de los módulos `CGI` y `DBI`.
- El CPAN (Comprehensive Perl Archive Network) organiza todos los módulos del mundo juntos y hace fácil la búsqueda a través de ellos.
- Los módulos `Inline` permiten a Perl ejecutar código de otros lenguajes de programación, tal como si fuera Perl nativo. Por ejemplo, `Inline::Python` permite a los programadores utilizar objetos de Python tal y como si estuvieran utilizando objetos de Perl.
- En Perl existen muchos conceptos y sintaxis de diferentes lenguajes de programación. Siempre es agradable poder observar algo familiar cuando se está aprendiendo algo nuevo. Aún los programadores novatos pueden beneficiarse de esto si pueden reconocer el parecido de Perl con el Inglés hablado (Beppu, 2002).

Un ejemplo sobresaliente del papel que ha jugado Perl en bioinformática, es cuando permitió a los científicos del Proyecto Genoma Humano el intercambiar datos y comparar los resultados que se estaban produciendo en 2 diferentes centros de secuenciamiento (Stein, 1996).

Existe una gran variedad de aplicaciones bioinformáticas desarrolladas en Perl, algunos ejemplos se encuentran en la siguiente tabla:

Tabla 3. Aplicaciones bioinformáticas desarrolladas en Perl

Aplicación	Descripción
MULTICLUSTAL	Automatiza el proceso de escoger los parámetros de alineamiento para *Clustal W*, con objeto de generar alineamientos de alta calidad (Yuan *et al*., 1999).
Oliz	Busca oligonucleótidos específicos a genes para experimentos de microarreglos (Chen *et al*., 2002).
Pise	Genera interfaces de Web a partir de programas de biología molecular (Letondal, 2001).
Swissknife	Permite extraer o modificar registros dentro de archivos con formato SWISS-PROT (Hermjakob *et al*., 1999).
Virtual PCR	Predice productos de PCR[4] a partir de iniciadores introducidos por el usuario (Lexa *et al*., 2001).
WebPHYLIP	Interfaz de Web para *PHYLIP*, permite realizar análisis filogenéticos a través de Internet (Lim *et al*., 1999).

En 1995 se formó la Open Bioinformatics Foundation (OBF) por un grupo de auto-denominados hackers[5] de Perl, con el objeto de reunir recursos para el desarrollo de software bioinformático. Esta fundación cuenta actualmente con los siguientes proyectos: BioPerl, BioPython, BioJava, BioCORBA y BioDAS. Dentro de estos, BioPerl es el más antiguo y utilizado, y por buenas razones. Es ciertamente el más maduro, posee las características más útiles y la comunidad de desarrollo más grande. La documentación de la estructura de BioPerl (aproximadamente 60 niveles, ~400 módulos) es manejada en un diseño extremadamente bien logrado que hace fácil examinar la funcionalidad del proyecto. La documentación para cada módulo es bastante completa y contiene descripciones cortas del módulo, sus dependencias, sus herencias y ejemplos de código para cada método (Mangalam, 2002).

[4] Polymerase Chain Reaction, técnica mediante la cual se puede amplificar DNA utilizando primers de oligonucleótidos y DNA polimerasa.

[5] La utilización del termino "hacker" para hacer referencia a un "violador de seguridad" es una confusión por parte de los medios de comunicación. El verdadero significado es el de: "alguien que ama la programación y disfruta el ser listo en eso" (Stallman, 1999).

2. Justificación

El análisis de secuencias es una de las metodologías más utilizadas en bioinformática y recientemente en biología molecular, por lo que es importante el desarrollo de herramientas computacionales adecuadas y eficientes para llevar a cabo el trabajo.

Hoy en día es posible realizar muchos de estos análisis mediante herramientas en Internet que facilitan la utilización y aprendizaje de estas metodologías, esto es a través de interfaces sencillas para los usuarios nuevos y al mismo tiempo poderosas para los usuarios avanzados. La mayoría de estas herramientas utilizan el lenguaje de programación Perl debido a su alta eficiencia para el procesamiento de textos y desarrollo de aplicaciones Web.

La base de datos PROSITE es una de las más conocidas y utilizadas para la identificación de dominios funcionales en secuencias de proteínas. Existen algunas herramientas que realizan búsquedas dentro de esta base de datos, desafortunadamente estas búsquedas están limitadas a que el usuario introduzca secuencias de proteína únicamente, excluyendo la posibilidad de hacer búsquedas a partir de secuencias de nucleótidos, las cuales son más usuales de obtener en los experimentos de laboratorio.

Resulta necesario desarrollar una aplicación que permita a los usuarios realizar búsquedas en la base de datos a partir de ambos tipos de secuencias (nucleótidos y/o proteína), para así poder obtener un mayor beneficio tanto de la base de datos como de las secuencias obtenidas en el laboratorio.

3. Objetivos

3.1 Objetivo General

Desarrollar una interfaz de Web para el análisis de secuencias de nucleótidos y/o aminoácidos que utilice la base de datos PROSITE para la búsqueda de dominios proteínicos conocidos en ellas.

3.2 Objetivos Particulares

- Elaborar un diseño de software adecuado para facilitar su administración y utilización.
- Utilizar para su implementación el lenguaje de programación Perl y evaluar su eficiencia.
- Evaluar la facilidad de utilización de la interfaz y de interpretación de sus resultados.
- Determinar la veracidad de los resultados proporcionados por la aplicación.

4. Metodología

4.1 Diseño de ProSA

ProSA es una abreviatura para "Protein Sequence Analyzer". La idea general de esta aplicación es que sea una interfaz de Web que permita a los usuarios realizar análisis de secuencias de nucleótidos y/o aminoácidos mediante búsquedas en la base de datos PROSITE, con la finalidad de poder predecir a que posible familia de proteínas pertenece una secuencia obtenida en el laboratorio.

Para llevar a cabo la implementación de esta aplicación, se requirió el planteamiento del siguiente diseño, el cual consta de 3 partes principales:

1. Un script que actualice regularmente la base de datos PROSITE.
2. Una interfaz de Web para el usuario.
3. Una aplicación CGI que realice el análisis e imprima los resultados al usuario.

A continuación se describen con mayor detalle cada una de las partes del diseño.

4.1.1 Script para actualizar la base de datos PROSITE

La base de datos PROSITE se encuentra como un archivo de texto. Lamentablemente es un archivo demasiado grande (~8 Mb y creciendo) como para accesar y extraer los registros de nuestro interés de manera remota. Además, la base de datos es actualizada aproximadamente cada 2 semanas, dentro de estas actualizaciones puede haber inserción de nuevos registros, corrección y/o eliminación de registros existentes.

Cabe recordar que la base de datos PROSITE contiene registros de patrones y perfiles. Debido al objetivo de la interfaz, solo se necesitan los registros pertenecientes a patrones. Además, a pesar de que los patrones contenidos en la base de datos se encuentran en forma de una expresión regular, estas no son expresiones regulares compatibles con Perl (Sigrist *et al*., 2002; Tisdall, 2001).

Por tales motivos, fue necesario tener una copia local de la base de datos, que sea actualizada con regularidad, así como elaborar una segunda base de datos que contenga únicamente patrones, que ya estén convertidos a un formato compatible con la aplicación que se utilizará posteriormente para los análisis.

Para llevar a cabo estas funciones, el script debía contar con las siguientes características:

- Ejecutarse regularmente para asegurar que la versión local de la base de datos fuese actual o descargarla en caso de que ésta no exista en el sistema.
- Elaborar una bitácora de la actividad del script para poder consultar el estado de la base de datos.
- Mantener informado al administrador de la aplicación sobre la actividad periódica del script y su estado.
- Elaborar una segunda base de datos que contenga únicamente los registros pertenecientes a patrones.
- Convertir dichos patrones en expresiones regulares compatibles con Perl.

4.1.2 Interfaz de Web para el usuario

Los formularios HTML[1] son la interfaz de usuario que provee la entrada de datos a las aplicaciones CGI. Estos son utilizados primordialmente con dos propósitos: colectar datos y aceptar comandos. Algunos ejemplos de datos para colectar pueden ser: información de registro, información de pago o encuestas en línea. También se pueden colectar comandos a través de formularios, comandos tales como menús, casillas, listas y/o botones para controlar diversos aspectos de una aplicación. En muchos casos, los formularios contienen elementos para ambas cosas: colectar datos y controlar la aplicación.

Una gran ventaja de los formularios HTML es que se pueden utilizar para crear una interfaz para numerosos servicios (tales como bases de datos u otros servidores de información), que pueden ser accesados por cualquier cliente desde cualquier sistema operativo (Guelich *et al*., 2000).

Para lograr la funcionalidad deseada, la interfaz debía poseer las siguientes características:

- Poder describir de manera sencilla el objetivo de la aplicación.
- Poseer instrucciones sobre su utilización.
- Contar con los campos necesarios para el análisis por la aplicación CGI.
- Contar con un vínculo para contactar al autor en caso de cualquier anomalía o comentario.

4.1.3 Aplicación CGI para el análisis

La Common Gateway Interface (CGI) puede hacer mucho debido a que es muy sencilla. La CGI es una interfaz muy ligera; es lo mínimo que el servidor de Web necesita para permitir a programas externos crear páginas Web. Típicamente, cuando un servidor de Web obtiene una petición de una página estática, el servidor de Web localiza el archivo HTML correspondiente en su sistema de archivos. Cuando un servidor de Web obtiene una petición de una aplicación CGI, el servidor de Web ejecuta dicha aplicación como un programa aislado; el servidor envía a dicho programa algunos parámetros y obtiene su respuesta, la cual es regresada al cliente tal como si hubiera sido extraída de un archivo estático (Guelich *et al*., 2000). La siguiente figura simplifica la comprensión de este proceso:

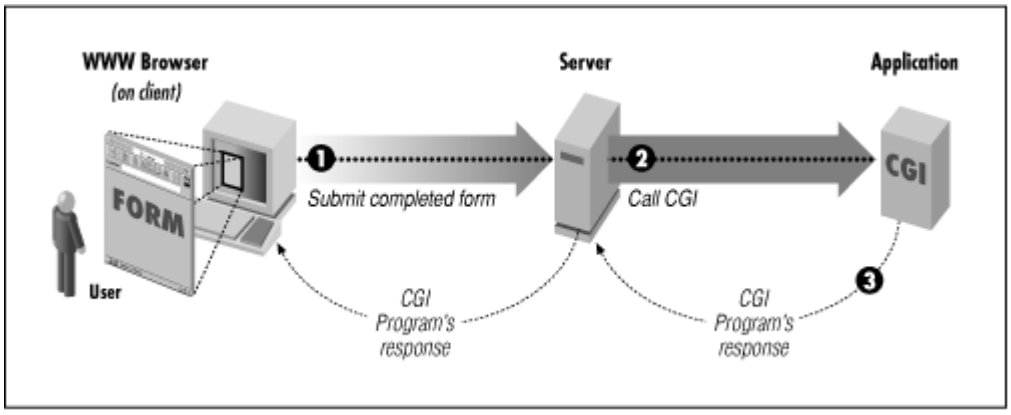


Figura 3. Modelo de ejecución de una aplicación CGI (Guelich *et al*., 2000)

Para llevar a cabo estas funciones y cumplir con el objetivo de la interfaz, la aplicación CGI debía poseer las siguientes características:

[1] Hypertext Markup Language, es el lenguaje estándar para diseñar documentos Web y especificar hipervínculos en ellos.

- Capturar los datos introducidos por el usuario e informarle sobre errores en caso de existir.
- Accesar la base de datos previamente elaborada para la aplicación.
- Realizar el análisis de acuerdo al tipo de secuencia que introduzca el usuario:
    - Secuencia de nucleótidos:
        - Eliminar de la secuencia los caracteres que no sean nucleótidos (A, T, G, C, U).
        - Convertir la secuencia en mRNA y obtener información útil sobre ésta, tal como: longitud, peso molecular, cantidad y porcentajes de A, U, G, C, y porcentajes de AU y GC.
        - Crear 6 marcos de lectura para su traducción a proteína.
        - Traducir cada marco de lectura de acuerdo al Código Genético (Elzanowski *et al*., 2000) seleccionado por el usuario.
        - Analizar cada marco de lectura:
            - Separando la secuencia por cada STOP (*) encontrado, obteniendo así subsecuencias de proteína.
            - Por cada subsecuencia de proteína:
                - Obtener su longitud y peso molecular.
                - Localizar patrones de PROSITE, así como almacenar su ubicación y descripción en caso de encontrarse.
    - Secuencia de aminoácidos:
        - Eliminar de la secuencia los caracteres que no sean aminoácidos (A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y, *).
        - Separar la secuencia por cada STOP (*) encontrado, obteniendo así subsecuencias de proteína.
        - Por cada subsecuencia de proteína:
            - Obtener su longitud y peso molecular.
            - Localizar patrones de PROSITE, así como almacenar su ubicación y descripción en caso de encontrarse.
- Imprimir los resultados en un formato congruente con la interfaz y legible para su interpretación.
- Calcular la duración del análisis con fines informativos.
- Elaborar una bitácora de la actividad de la aplicación para obtener información sobre su utilización.

4.2 Implementación de ProSA

Una vez elaborado el diseño de la aplicación, se procedió a su implementación para la cual se utilizó una PC con las siguientes características:

- Sistema operativo: FreeBSD 4.8 (Lehey, 2003)
- Interprete de Perl 5.8 (ultima versión estable[2] del lenguaje)
- Servidor de Web: Apache 1.3.27 (Laurie *et al*., 1999)

La aplicación se escribió en un editor de texto con apoyo de la bibliografía más conocida sobre Perl (Christiansen *et al*., 2003; Guelich *et al*., 2000; Schwartz *et al*., 2001; Spainhour *et al*., 2002; Srinivasan, 1997; Vromans, 2002; Wall *et al*., 2000; Wong, 1997), expresiones regulares (Friedl, 2002; Stubblebine, 2003) y la reciente bibliografía sobre Perl para bioinformática (Tisdall, 2001; Tisdall, 2003).

Cabe mencionar que la aplicación pudo haber sido desarrollada en cualquier otro sistema operativo (Linux, Mac OS ó Windows) que contara con un intérprete de Perl y un servidor de Web. La elección

[2] Versión que ha sido revisada exhaustivamente aprobando numerosas pruebas y por lo tanto es lo suficientemente madura para ser liberada para su utilización por el público en general.

tomada para este trabajo fue por decisión del autor debido a varias razones, dentro de las cuales destacan:

1. El robusto soporte de FreeBSD para el desarrollo de aplicaciones Web (Lehey, 2003).
2. La facilidad de administración y seguridad de Apache (Laurie *et al*., 1999).
3. FreeBSD, Perl y Apache son proyectos de software libre[3] que cuentan con una gran cantidad de documentación, soporte y usuarios, además de que son gratuitos.

4.3 Obtención de secuencias para análisis con ProSA

Para comprobar la eficiencia de la aplicación, se hizo una selección en GenBank (*http://www.ncbi.nlm.nih.gov/Genbank/*) de 10 secuencias con tablas de traducción diferentes. La siguiente tabla describe brevemente cada una de estas secuencias:

Tabla 4. Secuencias seleccionadas para análisis con ProSA

Secuencia	Número(s) de Acceso	Longitud (nucleótidos)	Tabla de Traducción	Referencia
Virus de inmunodeficiencia humana 1, genoma completo	NC_001802	9181	Standard (1)	Petropoulos (1997)
DNA mitocondrial de *Chelonia mydas*, secuencia completa	NC_000886	16497	Vertebrate Mitochondrial (2)	Kumazawa *et al*. (1999)
Mitocondria de *Saccharomyces cerevisiae*, genoma completo	NC_001224	85779	Yeast Mitochondrial (3)	*Saccharomyces* Genome Database (1999)
Mitocondria de *Acanthamoeba castellanii*, genoma completo	U12386	41591	Mold Mitochondrial, Protozoan Mitochondrial, Coelenterate Mitochondrial, Mycoplasma, Spiroplasma (4)	Burger *et al*. (1995)
Mitocondria de *Drosophila melanogaster*, genoma completo	NC_001709	19517	Invertebrate Mitochondrial (5)	Lewis *et al*. (1995)
mRNA para hemoglobina de *Tetrahymena pyriformis*	D13920	587	Ciliate Nuclear, Dasycladacean Nuclear, Hexamita Nuclear (6)	Takagi *et al*. (1993)
mRNA macronuclear para protein-cinasa nuclear putativa de *Euplotes octocarinatus* (gen npk 1)	AJ249683	1322	Euplotid Nuclear (10)	Tan *et al*. (2001)
Gen potenciador de la infectividad a macrófagos (mip) de *Legionella lytica* cepa LLAP-9, cds parciales	AF148986	598	Bacterial and Plant Plastid (11)	Adeleke *et al*. (2001)
Mitocondria de *Scenedesmus obliquus*, genoma completo	X17375 AJ271733 AJ272528 AJ277429 AJ400708	42781	*Scenedesmus obliquus* mitochondrial (22)	Kuck *et al*. (2000)
DNA mitocondrial de *Thraustochytrium aureum*, genoma parcial	AF288091	31570	*Thraustochytrium* mitochondrial code (23)	Burger *et al*. (2000)

Estas secuencias fueron almacenadas en archivos con formato GenBank (Leon *et al*., 2003). Para el análisis de cada secuencia, se copiaron las líneas correspondientes a los nucleótidos y se introdujeron en la interfaz de Web, seleccionando el tipo de secuencia y tabla de traducción correspondiente. Se utilizaron como clientes los navegadores Web: Mozilla, Konqueror e Internet Explorer para evaluar el formato de la interfaz y sus resultados.

[3] Software que es distribuido con la totalidad de su código fuente, por lo que puede ser modificado/adaptado por el usuario para satisfacer sus necesidades.

5. Resultados

5.1 Implementación de ProSA

El resultado de la implementación consistió en una serie de archivos jerarquizados de acuerdo a su función dentro de la aplicación. La siguiente figura resume el sistema de archivos obtenido:

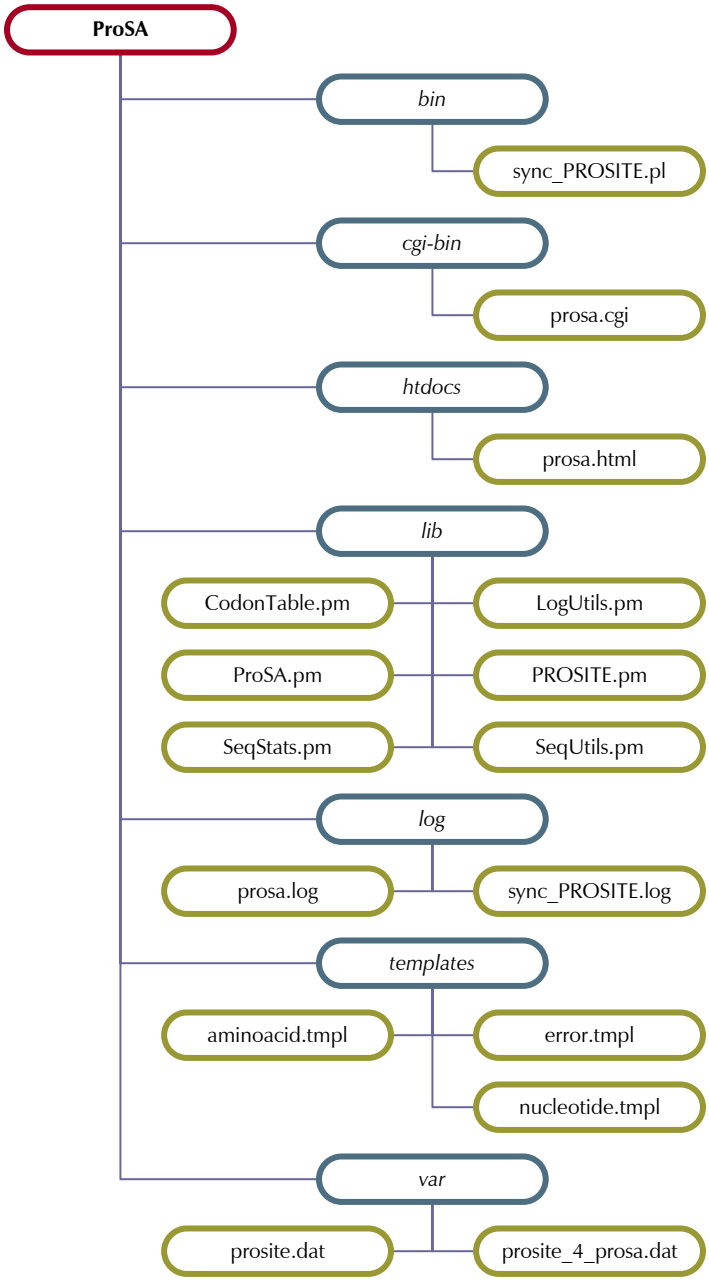


Figura 4. Jerarquía de archivos de ProSA

Este sistema de archivos se compone de directorios organizados de manera similar a la convención de los sistemas Unix. Esta elección es comúnmente utilizada para el desarrollo de aplicaciones, lo cual permite organizar los archivos de manera sencilla y eficiente (Gibas *et al*., 2001). La siguiente tabla describe el contenido de los directorios en la presente implementación:

Tabla 5. Contenido de los directorios de ProSA

Directorio	Contenido
bin	Script para actualizar la base de datos PROSITE
cgi-bin	Aplicación CGI que realiza el análisis
htdocs	Interfaz de Web para el usuario
lib	Módulos para el script de actualización y la aplicación CGI
log	Bitácoras de actividad
templates	Plantillas HTML para el despliegue de los resultados de la aplicación CGI
var	Archivos de la base de datos PROSITE

Para lograr algunas partes de la implementación, se recurrió a los siguientes módulos de Perl existentes: `CGI`, `Date::Calc`, `HTML::Template` y `LWP::Simple`. El primero forma parte de la distribución estándar de Perl, mientras que los demás fueron instalados de CPAN (*http://www.cpan.org/*). Todos los módulos contenidos en el directorio `lib` fueron elaborados por el autor para satisfacer las necesidades particulares de la implementación. El papel de estos módulos será descrito a lo largo de ésta sección.

A continuación se describen los resultados de la implementación para cada una de las partes propuestas en el diseño. En esta sección no se incluye el código fuente correspondiente a todas las funciones aquí mencionadas, por lo que en caso de quererlo examinar será necesario revisar el Apéndice B.

5.1.1 Script para actualizar la base de datos PROSITE

La implementación de este script (`sync_PROSITE.pl`) se llevó a cabo exitosamente, a continuación se muestran las soluciones planteadas para su diseño y sus resultados correspondientes.

5.1.1.1 Actualización de la base de datos

Para la actualización periódica de la base de datos se utilizó el módulo `LWP::Simple`, el cual permite realizar tareas a través de Internet, de ahí su nombre "Library for WWW Access in Perl" (Wong, 1997). Se utilizaron sus funciones `mirror()` y `status_message()` para sincronizar la base de datos local (`prosite.dat`) con la remota (*ftp://ftp.expasy.org/databases/prosite/release_with_updates/prosite.dat*) y para notificar sobre errores en caso de ocurrir.

5.1.1.2 Extracción de los registros pertenecientes a patrones

Se diseñó la función `parse_PROSITE()` del módulo `PROSITE` para extraer únicamente los registros pertenecientes a patrones y obtener de éstos los valores de los tipos de línea: ID, AC, DE y PA.

5.1.1.3 Conversión de los patrones de PROSITE en expresiones regulares de Perl

Mientras se extraían los patrones para cada uno de los registros, éstos eran convertidos en expresiones regulares de Perl utilizando la función `PROSITE_2_regexp()` del módulo `PROSITE`, dicha función se diseñó utilizando la descripción del lenguaje de patrones que se encuentra en el manual de usuario de PROSITE, la cual se muestra a continuación:

```
Las líneas PA (PAttern) contienen la definición de un patrón de PROSITE. Los patrones son descritos utilizando
las siguientes convenciones:

    · Se utilizan los códigos IUPAC de una sola letra para los aminoácidos.
    · El símbolo 'x' es utilizado para una posición donde cualquier aminoácido es aceptable.
    · Las ambigüedades son indicadas enlistando dentro de corchetes '[ ]' los aminoácidos aceptables para una
      posición determinada. Por ejemplo: [ALT] significa Ala ó Leu ó Thr.
    · Las ambigüedades también se encuentran indicadas enlistando dentro de un par de llaves '{ }' los
      aminoácidos que no son aceptables en una posición determinada. Por ejemplo: {AM} significa cualquier
      aminoácido excepto Ala y Met.
    · Cada elemento en un patrón es separado de su vecino mediante un guión '-'.
    · La repetición de un elemento del patrón está indicada mediante un número o un rango numérico dentro de
      paréntesis. Ejemplos: x(3) corresponde a x-x-x; x(2,4) corresponde a x-x ó x-x-x ó x-x-x-x.
    · Cuando un patrón está restringido al extremo C- o N- de una secuencia, ese patrón comienza con una llave
      '<' o termina con un '>' respectivamente.
    · Un punto termina el patrón.
```

5.1.1.4 Elaboración de una segunda base de datos

Una vez obtenidos los valores de identificación, número de acceso, descripción corta, patrón y expresión regular equivalente para todos los registros, éstos fueron almacenados en una segunda base de datos (`prosite_4_prosa.dat`) mediante la función `make_PROSITE_4_ProSA()` del módulo `PROSITE`. El archivo resultante tuvo un tamaño considerablemente menor (~200 Kb). A continuación se muestra su formato:

```
ASN_GLYCOSYLATION||PS00001||N-glycosylation site||N-{P}-[ST]-{P}||N[^P][ST][^P]
PKC_PHOSPHO_SITE||PS00005||Protein kinase C phosphorylation site||[ST]-x-[RK]||[ST].[RK]
CK2_PHOSPHO_SITE||PS00006||Casein kinase II phosphorylation site||[ST]-x(2)-[DE]||[ST].{2}[DE]
MYRISTYL||PS00008||N-myristoylation site||G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}||G[^EDRKHPFYW].{2}[STAGCN][^P]
AMIDATION||PS00009||Amidation site||x-G-[RK]-[RK]||.G[RK][RK]
```

5.1.1.5 Elaboración de una bitácora, ejecución periódica y notificación sobre la actividad del script

Para la elaboración de una bitácora de la actividad del script se diseñó la función `write_log()` del módulo `LogUtils`, la cual se encarga de escribir en un archivo (`sync_PROSITE.log`) los mensajes del script que poseen el siguiente formato:

```
[Mon Nov 10 00:00:04 2003] [notice] La sincronización inició.
[Mon Nov 10 00:01:19 2003] [notice] Se descargó la base de datos PROSITE.
[Mon Nov 10 00:01:21 2003] [notice] Se creó el archivo destino: /home/web-biol/var/prosite/prosite_4_prosa.dat
[Mon Nov 10 00:01:21 2003] [notice] La sincronización finalizó con éxito.
```

La ejecución periódica del script se logró utilizando la herramienta *cron* del sistema operativo (Peek *et al*., 1997), para lo cual se añadieron las siguientes líneas al archivo `crontab`:

```
MAILTO=web-biol@campus.iztacala.unam.mx
0 0 * * 1     /home/web-biol/bin/sync_PROSITE.pl
```

La primera línea solicita a *cron* que envíe un correo a la dirección especificada cada vez que se ejecute el script, con lo cual se notifica al administrador de la aplicación sobre la actividad del script y el estado de la base de datos. Dada la constante actualización de la base de datos PROSITE, esta ejecución se programó en la segunda línea solicitándose para todos los días Lunes a las 0 hrs. A continuación se muestra un ejemplo del mensaje de correo que se recibe durante cada ejecución, el cuerpo del mensaje contiene los mensajes del script enviados por la función `write_stdout()` del módulo `LogUtils`:

```
Date: Mon, 10 Nov 2003 00:00:04 -0600 (CST)
From: Cron Daemon <root@campus.iztacala.unam.mx>
To: web-biol@campus.iztacala.unam.mx
Subject: Cron <web-biol@campus> /home/web-biol/bin/sync_PROSITE.pl

[Mon Nov 10 00:00:04 2003] [notice] La sincronización inició.
```

```
[Mon Nov 10 00:01:19 2003] [notice] Se descargó la base de datos PROSITE.
[Mon Nov 10 00:01:21 2003] [notice] Se creó el archivo destino: /home/web-biol/var/prosite/prosite_4_prosa.dat
[Mon Nov 10 00:01:21 2003] [notice] La sincronización finalizó con éxito.
```

5.1.2 Interfaz de Web para el usuario

La implementación de la interfaz (`prosa.html`) se llevó a cabo exitosamente. La siguiente figura muestra la pantalla de un navegador Web con el documento HTML elaborado:

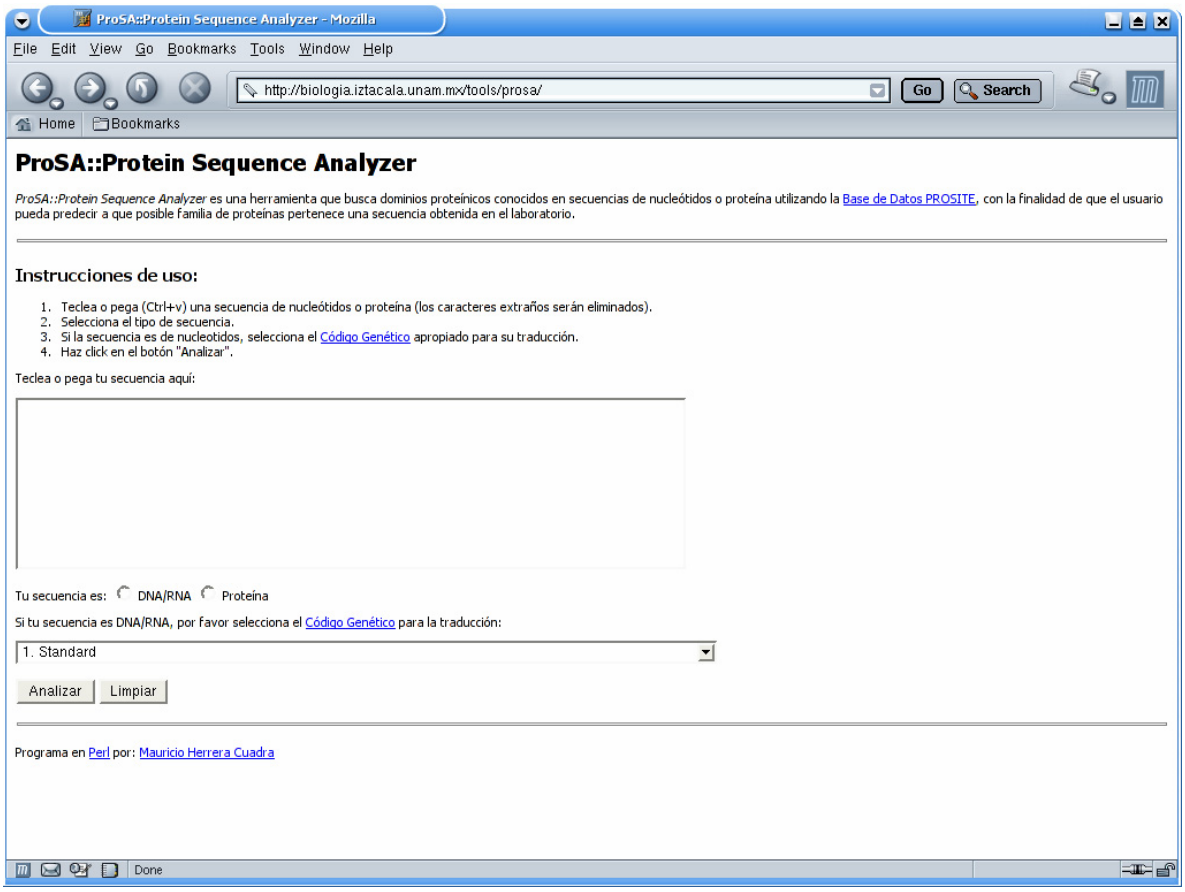


Figura 5. Apariencia de la interfaz de ProSA

Como podemos observar, la interfaz cuenta con todos los elementos planteados en su diseño. Ésta posee una breve descripción del objetivo de la aplicación, instrucciones sobre su utilización, los campos necesarios para el análisis, y un vínculo para contactar al autor en caso de cualquier anomalía o comentario.

Se aprovecharon las ventajas que brinda el lenguaje HTML para incluir vínculos adicionales hacia el sitio de la base de datos PROSITE (*http://www.expasy.org/prosite/*), el sitio del NCBI donde se encuentra una explicación detallada sobre los Códigos Genéticos (*http://www.ncbi.nlm.nih.gov/htbin-post/Taxonomy/wprintgc?mode=c*) y el sitio de Perl (*http://www.perl.com/*) en caso de que los usuarios deseen obtener mayor información.

5.1.3 Aplicación CGI para el análisis

La implementación de la aplicación CGI (`prosa.cgi`) se llevó a cabo exitosamente, a continuación se muestran las soluciones planteadas para su diseño y sus resultados correspondientes.

5.1.3.1 Captura de datos e información de errores

Para realizar la captura de los datos introducidos por el usuario se utilizó el módulo `CGI`, el cual se ha convertido en la herramienta estándar para la creación de aplicaciones CGI en Perl (Guelich *et al*., 2000).

Para informar al usuario en caso de ocurrir algún error (al introducir datos o al seleccionar los parámetros para el análisis) se utilizó la función `printError()` de la aplicación CGI (`prosa.cgi`).

5.1.3.2 Acceso a la base de datos

El acceso a la segunda base de datos (`prosite_4_prosa.dat`) se logró mediante la función `load_PROSITE()` del módulo `PROSITE`, la cual coloca en memoria todos los registros de PROSITE pertenecientes a patrones que fueron previamente preparados para la aplicación.

5.1.3.3 Análisis según el tipo de secuencia

5.1.3.3.1 Secuencias de nucleótidos

Para la eliminación de los caracteres en la secuencia que no fueran nucleótidos (A, T, G, C, U) y la conversión de la secuencia en mRNA para su traducción, se utilizaron las funciones `sanitize_seq()` y `DNA_2_mRNA()` del módulo `SeqUtils`.

La obtención de información útil sobre la secuencia, tal como: longitud, peso molecular, cantidad y porcentajes de A, U, G, C, y porcentajes de AU y GC se logró mediante las funciones `length()` de Perl y `seq_mw()`, `n_number()` y `n_percent()` del módulo `SeqStats` respectivamente.

La creación de 6 marcos de lectura y la traducción a proteína se llevó a cabo con las funciones `make_orfs()` y `translate()` del módulo `SeqUtils`. La tabla de traducción correspondiente al Código Genético (Elzanowski *et al*., 2000) seleccionado por el usuario se obtuvo mediante la función `get_trans_table()` del módulo `CodonTable`. En el Apéndice A se encuentran las tablas de traducción correspondientes a todos los Códigos Genéticos utilizados.

Para analizar cada marco de lectura generado se utilizó la función `analyze_protein()` del módulo `ProSA`, la cual emplea a su vez las funciones: `split_by_stops()` del módulo `SeqUtils` para separar la secuencia por cada STOP (*) encontrado; `length()` de Perl y `seq_mw()` del módulo `SeqStats` para obtener la longitud y peso molecular de cada subsecuencia producida por la separación en cada STOP; y finalmente `search_PROSITE()` del módulo `PROSITE` para buscar en cada subsecuencia los patrones de PROSITE, así como almacenar su ubicación y descripción en caso de encontrarse.

5.1.3.3.2 Secuencias de proteína

Para la eliminación de los caracteres en la secuencia que no fueran aminoácidos (A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y, *) se empleó la función `sanitize_seq()` del módulo `SeqUtils`.

Para analizar la secuencia se utilizó la función `analyze_protein()` del módulo `ProSA`, la cual lleva a cabo las mismas funciones descritas anteriormente para el análisis de secuencias de nucleótidos.

5.1.3.4 Impresión de los resultados

La impresión de los resultados en un formato congruente con la interfaz y legible para su interpretación fue lograda con la ayuda del módulo `HTML::Template`, el cual soporta plantillas HTML complejas para el despliegue de resultados de aplicaciones CGI (Guelich *et al.*, 2000). Las plantillas elaboradas para la interfaz fueron: `aminoacid.tmpl`, `error.tmpl` y `nucleotide.tmpl`, las cuales son empleadas por la aplicación CGI según sea el caso (análisis de secuencias de proteína, despliegue de errores y análisis de secuencias de nucleótidos, respectivamente).

5.1.3.5 Cálculo de la duración del análisis

Para calcular la duración del análisis se emplearon las funciones `Today_and_Now()` y `Delta_DHMS()` del módulo `Date::Calc`, la primera para capturar los momentos de inicio y final del análisis, y la segunda para calcular la diferencia entre ambos tiempos.

5.1.3.6 Elaboración de una bitácora

Para la elaboración de una bitácora de la actividad de la aplicación se utilizó la función `write_log()` del módulo `LogUtils`, la cual se encarga de escribir en un archivo (`prosa.log`) los mensajes de la aplicación que poseen el siguiente formato:

```
[Sat Nov 15 18:10:22 2003] [notice] [client 127.0.0.1] ProSA/1.0 inició.
[Sat Nov 15 18:10:22 2003] [notice] [client 127.0.0.1] La secuencia introducida fué:
        1 agccgtttta gggcttgcaa tgtctactgc aatggctgct gatgcattag ttacagacaa
       61 agacaaattg tcttacagta ttggtgcaga tttagggaaa aattttaaag gtcaaggcat
      121 cgatatcaat cctgaagcat tagccaaagg aatgcaagat ggaatgtctg gcgctcaatt
      181 gattttaact gaacaacaaa tgaaagatgt tttaaacaaa tttcaaaaag atttaatggc
      241 taagcgtagc gcagaattta ataaaaaagc tgaagaaaac aaatctaaag gcgaagcgtt
      301 tttaagttct aataaagcaa aaactggtgt agtagtatta ccaagcggct tgcaatataa
      361 aattcttgaa gccggtactg gtgctaagcc aggaaaagca gatactgtaa ctgttgatta
      421 cactggtact ttgatcgacg gtactgtatt tgatagcact caaaagactg gtaaaccagc
      481 tacattccaa gtatcacaag ttattccagg ctggactgaa gcattacaat taatgcctgc
      541 tggttctact tgggaagttt ttgttcctgc tgatctagct tacggtccac gtagtgtt
[Sat Nov 15 18:10:22 2003] [notice] [client 127.0.0.1] El tipo de secuencia fué: nucleotide
[Sat Nov 15 18:10:22 2003] [notice] [client 127.0.0.1] El Código Genético seleccionado fué: Bacterial and
Plant Plastid
[Sat Nov 15 18:10:22 2003] [notice] [client 127.0.0.1] El análisis comenzó.
[Sat Nov 15 18:10:26 2003] [notice] [client 127.0.0.1] El análisis terminó.
[Sat Nov 15 18:10:26 2003] [notice] [client 127.0.0.1] La duración del análisis fué: 00:00:04
[Sat Nov 15 18:10:26 2003] [notice] [client 127.0.0.1] ProSA/1.0 finalizó con éxito.
```

5.2 Ejemplo de utilización de ProSA

A continuación se describirá paso a paso y con ayuda de figuras la realización de un análisis con ProSA.

5.2.1 Introducción de una secuencia y selección de parámetros

- Introducir la secuencia que se desea analizar en la caja de texto más grande que aparece en el formulario HTML (**Paso 1**).
- Seleccionar un tipo de secuencia para el análisis (**Paso 2**).
- Seleccionar un Código Genético para la traducción en caso de que la secuencia sea de nucleótidos (**Paso 3**).
- Hacer clic en el botón "*Analizar*" (**Paso 4**).

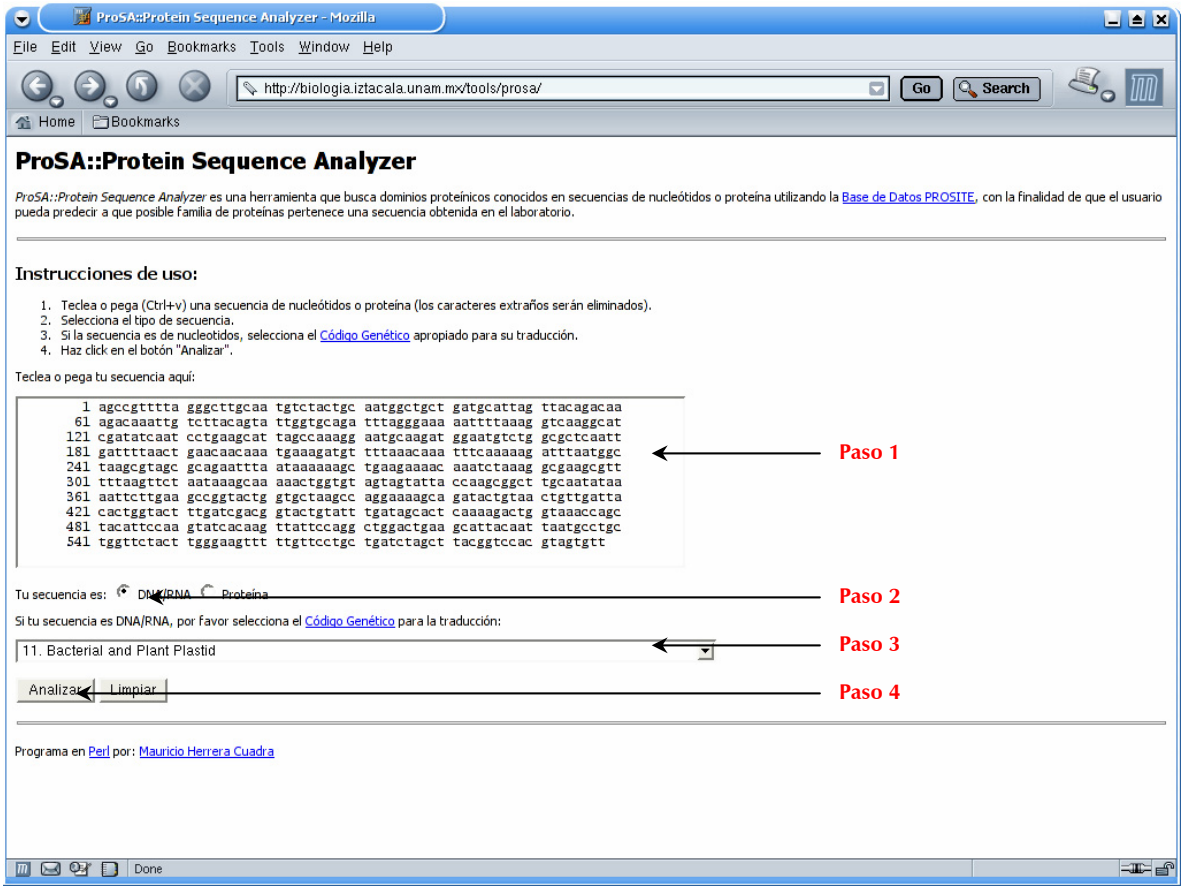


Figura 6. Ejemplo de llenado del formulario de ProSA

5.2.2 Despliegue de los resultados del análisis

En la siguiente figura se muestran los resultados del análisis de la secuencia introducida. Se pueden observar datos como: secuencia introducida, duración del análisis, secuencia de mRNA utilizada para la traducción, número de nucleótidos, peso molecular, número y porcentaje de Adeninas, Uracilos, Guaninas, Citosinas, porcentaje de Adenina-Uracilo y Guanina-Citosina, Código Genético utilizado para la traducción, secuencia del 1er marco de lectura, número de subsecuencias, longitud y peso molecular de cada una de ellas, y los datos más importantes obtenidos con esta aplicación: los patrones de PROSITE que se localizaron en cada subsecuencia, así como su posición, fragmento concordante, patrón de búsqueda y expresión regular equivalente.

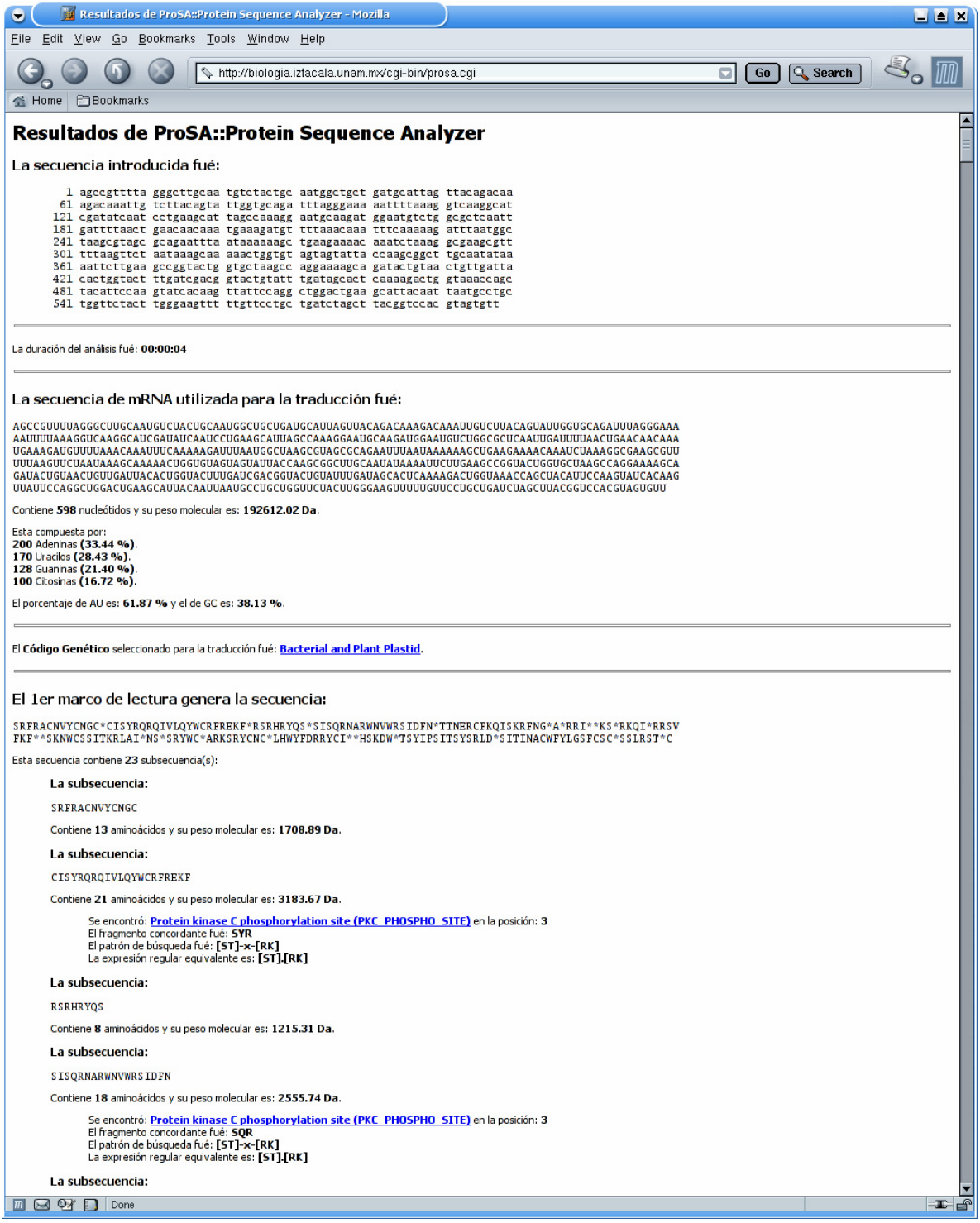


Figura 7. Página con los resultados del análisis realizado por ProSA

Nuevamente se aprovecharon las ventajas del lenguaje HTML para incluir vínculos en la página de resultados del análisis. El primero es hacia el sitio del NCBI donde se encuentra la información referente a los Códigos Genéticos. Para cada análisis de secuencias de nucleótidos éste vínculo se estará apuntando hacia la sección correspondiente al Código Genético seleccionado por el usuario. Los

vínculos restantes son hacia la documentación correspondiente a cada patrón de PROSITE encontrado en las subsecuencias de proteína.

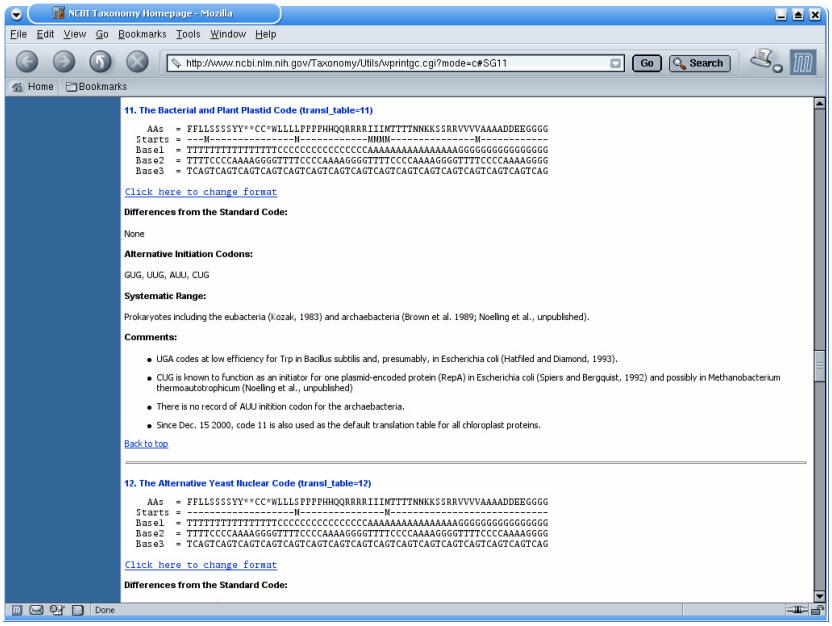


Figura 8. Página con la explicación del Código Genético utilizado para la traducción

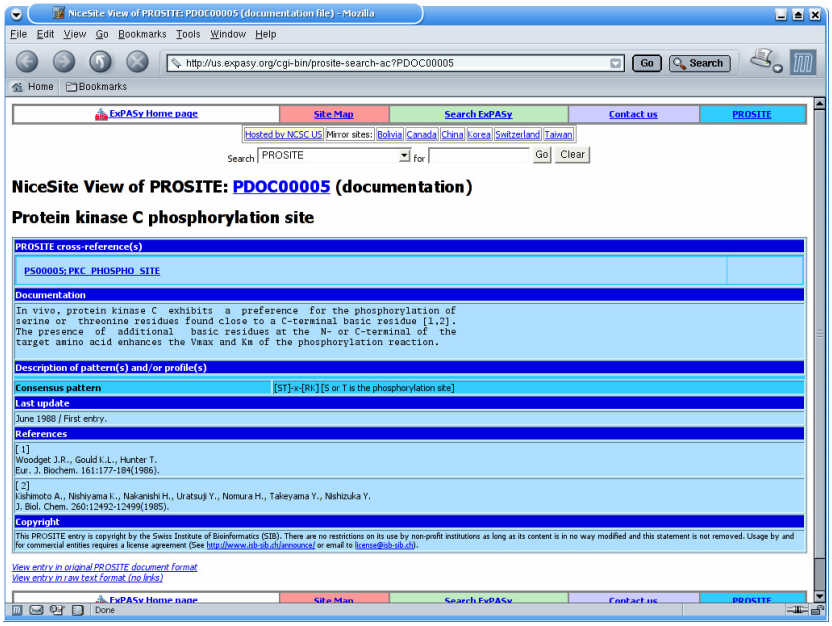


Figura 9. Página con la documentación de uno de los patrones encontrados en el análisis

5.3 Análisis con ProSA

Una vez realizado el análisis de las secuencias seleccionadas en GenBank, se procedió a la interpretación de los resultados obtenidos por la aplicación CGI. Los resultados completos fueron almacenados en

archivos HTML. Se utilizó como nombre para cada archivo el número de acceso correspondiente en GenBank para cada secuencia.

La cantidad de información obtenida en los 10 análisis fue demasiado extensa (~14 Mb), por lo que en esta sección solo se reproducirán y describirán brevemente los resultados obtenidos para una de las secuencias seleccionadas. Al final de la sección se presentan tablas con los resultados simplificados para los análisis de las 10 secuencias.

5.3.1 Ejemplo de interpretación de resultados de ProSA

A continuación se muestran los resultados abreviados para la secuencia del Virus de inmunodeficiencia humana 1, genoma completo y su interpretación. Para señalar las concordancias correspondientes al bloque en turno se intercalan fragmentos de los archivos obtenidos en GenBank. Se emplean las siguientes convenciones para su representación:

`Ancho Fijo`
Resultados del análisis y ausencia de concordancias con lo publicado en GenBank.

`Ancho Fijo + Cursiva`
Concordancias con lo publicado en GenBank pero un marco de lectura diferente.

`Ancho Fijo + Negrita`
Concordancias con lo publicado en GenBank.

5.3.1.1 Virus de inmunodeficiencia humana 1, genoma completo

```
Resultados de ProSA::Protein Sequence Analyzer

La secuencia introducida fué:

       1 ggtctctctg gttagaccag atctgagcct gggagctctc tggctaacta gggaacccac
      61 tgcttaagcc tcaataaagc ttgccttgag tgcttcaagt agtgtgtgcc cgtctgttgt
     121 gtgactctgg taactagaga tccctcagac ccttttagtc agtgtggaaa atctctagca
     181 gtggcgcccg aacagggacc tgaaagcgaa agggaaacca gaggagctct ctcgacgcag
     241 gactcggctt gctgaagcgc gcacggcaag aggcgagggg cggcgactgg tgagtacgcc
     301 aaaaattttg actagcggag gctagaagga gagagatggg tgcgagagcg tcagtattaa
     361 gcgggggaga attagatcga tgggaaaaaa ttcggttaag gccaggggga aagaaaaaat
     421 ataaattaaa acatatagta tgggcaagca gggagctaga acgattcgca gttaatcctg
     481 gcctgttaga aacatcagaa ggctgtagac aaatactggg acagctacaa ccatcccttc
     541 agacaggatc agaagaactt agatcattat ataatacagt agcaaccctc tattgtgtgc
     601 atcaaaggat agagataaaa gacaccaagg aagctttaga caagatagag gaagagcaaa
     661 acaaaagtaa gaaaaaagca cagcaagcag cagctgacac aggacacagc aatcaggtca
     721 gccaaaatta ccctatagtg cagaacatcc aggggcaaat ggtacatcag gccatatcac
     781 ctagaacttt aaatgcatgg gtaaaagtag tagaagagaa ggctttcagc ccagaagtga
     841 tacccatgtt ttcagcatta tcagaaggag ccaccccaca agatttaaac accatgctaa
     901 acacagtggg gggacatcaa gcagccatgc aaatgttaaa agagaccatc aatgaggaag
     961 ctgcagaatg ggatagagtg catccagtgc atgcagggcc tattgcacca ggccagatga
    1021 gagaaccaag gggaagtgac atagcaggaa ctactagtac ccttcaggaa caaataggat
    1081 ggatgacaaa taatccacct atcccagtag gagaaattta taaaagatgg ataatcctgg
    1141 gattaaataa aatagtaaga atgtatagcc ctaccagcat tctggacata agacaaggac
    1201 caaaggaacc ctttagagac tatgtagacc ggttctataa aactctaaga gccgagcaag
    1261 cttcacagga ggtaaaaaat tggatgacag aaaccttgtt ggtccaaaat gcgaacccag
    1321 attgtaagac tattttaaaa gcattgggac cagcggctac actagaagaa atgatgacag
    1381 catgtcaggg agtaggagga cccggccata aggcaagagt tttggctgaa gcaatgagcc
    1441 aagtaacaaa ttcagctacc ataatgatgc agagaggcaa ttttaggaac caaagaaaga
    1501 ttgttaagtg tttcaattgt ggcaaagaag ggcacacagc cagaaattgc agggccccta
    1561 ggaaaaaggg ctgttggaaa tgtggaaagg aaggacacca aatgaaagat tgtactgaga
    1621 gacaggctaa ttttttaggg aagatctggc cttcctacaa gggaaggcca gggaattttc
    1681 ttcagagcag accagagcca acagccccac cagaagagag cttcaggtct ggggtagaga
    1741 caacaactcc ccctcagaag caggagccga tagacaagga actgtatcct ttaacttccc
```

1801 tcaggtcact ctttggcaac gacccctcgt cacaataaag ataggggggc aactaaagga
1861 agctctatta gatacaggag cagatgatac agtattagaa gaaatgagtt tgccaggaag
1921 atggaaacca aaaatgatag ggggaattgg aggttttatc aaagtaagac agtatgatca
1981 gatactcata gaaatctgtg gacataaagc tataggtaca gtattagtag gacctacacc
2041 tgtcaacata attggaagaa atctgttgac tcagattggt tgcactttaa attttcccat
2101 tagccctatt gagactgtac cagtaaaatt aaagccagga atggatggcc caaaagttaa
2161 acaatggcca ttgacagaag aaaaaataaa agcattagta gaaatttgta cagagatgga
2221 aaaggaaggg aaaatttcaa aaattgggcc tgaaaatcca tacaatactc cagtatttgc
2281 cataaagaaa aaagacagta ctaaatggag aaaattagta gatttcagag aacttaataa
2341 gagaactcaa gacttctggg aagttcaatt aggaatacca catcccgcag ggttaaaaaa
2401 gaaaaaatca gtaacagtac tggatgtggg tgatgcatat ttttcagttc ccttagatga
2461 agacttcagg aagtatactg catttaccat acctagtata aacaatgaga caccagggat
2521 tagatatcag tacaatgtgc ttccacaggg atggaaagga tcaccagcaa tattccaaag
2581 tagcatgaca aaaatcttag agccttttag aaaacaaaat ccagacatag ttatctatca
2641 atacatggat gatttgtatg taggatctga cttagaaata gggcagcata gaacaaaaat
2701 agaggagctg agacaacatc tgttgaggtg gggacttacc acaccagaca aaaaacatca
2761 gaaagaacct ccattccttt ggatgggtta tgaactccat cctgataaat ggacagtaca
2821 gcctatagtg ctgccagaaa aagacagctg gactgtcaat gacatacaga agttagtggg
2881 gaaattgaat tgggcaagtc agatttaccc agggattaaa gtaaggcaat tatgtaaact
2941 ccttagagga accaaagcac taacagaagt aataccacta acagaagaag cagagctaga
3001 actggcagaa aacagagaga ttctaaaaga accagtacat ggagtgtatt atgacccatc
3061 aaaagactta atagcagaaa tacagaagca ggggcaaggc caatggacat atcaaattta
3121 tcaagagcca tttaaaaatc tgaaaacagg aaaatatgca agaatgaggg gtgcccacac
3181 taatgatgta aaacaattaa cagaggcagt gcaaaaaata accacagaaa gcatagtaat
3241 atggggaaag actcctaaat ttaaactgcc catacaaaag gaaacatggg aaacatggtg
3301 gacagagtat tggcaagcca cctggattcc tgagtgggag tttgttaata cccctccctt
3361 agtgaaatta tggtaccagt tagagaaaga acccatagta ggagcagaaa ccttctatgt
3421 agatggggca gctaacaggg agactaaatt aggaaaagca ggatatgtta ctaatagagg
3481 aagacaaaaa gttgtcaccc taactgacac aacaaatcag aagactgagt tacaagcaat
3541 ttatctagct ttgcaggatt cgggattaga agtaaacata gtaacagact cacaatatgc
3601 attaggaatc attcaagcac aaccagatca aagtgaatca gagttagtca atcaaataat
3661 agagcagtta ataaaaaagg aaaaggtcta tctggcatgg gtaccagcac acaaaggaat
3721 tggaggaaat gaacaagtag ataaattagt cagtgctgga atcaggaaag tactattttt
3781 agatggaata gataaggccc aagatgaaca tgagaaatat cacagtaatt ggagagcaat
3841 ggctagtgat tttaacctgc cacctgtagt agcaaaagaa atagtagcca gctgtgataa
3901 atgtcagcta aaaggagaag ccatgcatgg acaagtagac tgtagtccag gaatatggca
3961 actagattgt acacatttag aaggaaaagt tatcctggta gcagttcatg tagccagtgg
4021 atatatagaa gcagaagtta ttccagcaga aacagggcag gaaacagcat attttctttt
4081 aaaattagca ggaagatggc cagtaaaaac aatacatact gacaatggca gcaatttcac
4141 cggtgctacg gttagggccg cctgttggtg ggcgggaatc aagcaggaat ttggaattcc
4201 ctacaatccc caaagtcaag gagtagtaga atctatgaat aaagaattaa agaaaattat
4261 aggacaggta agagatcagg ctgaacatct taagacagca gtacaaatgg cagtattcat
4321 ccacaatttt aaaagaaaag gggggattgg ggggtacagt gcaggggaaa gaatagtaga
4381 cataatagca acagacatac aaactaaaga attacaaaaa caaattacaa aaattcaaaa
4441 ttttcgggtt tattacaggg acagcagaaa tccactttgg aaaggaccag caaagctcct
4501 ctggaaaggt gaaggggcag tagtaataca agataatagt gacataaaag tagtgccaag
4561 aagaaaagca aagatcatta gggattatgg aaaacagatg gcaggtgatg attgtgtggc
4621 aagtagacag gatgaggatt agaacatgga aaagtttagt aaaacaccat atgtatgttt
4681 cagggaaagc taggggatgg ttttatagac atcactatga aagccctcat ccaagaataa
4741 gttcagaagt acacatccca ctaggggatg ctagattggt aataacaaca tattggggtc
4801 tgcatacagg agaaagagac tggcatttgg gtcagggagt ctccatagaa tggaggaaaa
4861 agagatatag cacacaagta gaccctgaac tagcagacca actaattcat ctgtattact
4921 ttgactgttt ttcagactct gctataagaa aggccttatt aggacacata gttagcccta
4981 ggtgtgaata tcaagcagga cataacaagg taggatctct acaatacttg gcactagcag
5041 cattaataac accaaaaaag ataaagccac ctttgcctag tgttacgaaa ctgacagagg
5101 atagatggaa caagccccag aagaccaagg gccacagagg gagccacaca atgaatggac
5161 actagagctt ttagaggagc ttaagaatga agctgttaga cattttccta ggatttggct
5221 ccatggctta gggcaacata tctatgaaac ttatggggat acttgggcag gagtggaagc
5281 cataataaga attctgcaac aactgctgtt tatccatttt cagaattggg tgtcgacata
5341 gcagaatagg cgttactcga cagaggagag caagaaatgg agccagtaga tcctagacta
5401 gagccctgga agcatccagg aagtcagcct aaaactgctt gtaccaattg ctattgtaaa
5461 aagtgttgct ttcattgcca agtttgtttc ataacaaaag ccttaggcat ctcctatggc
5521 aggaagaagc ggagacagcg acgaagagct catcagaaca gtcagactca tcaagcttct
5581 ctatcaaagc agtaagtagt acatgtaatg caacctatac caatagtagc aatagtagca
5641 ttagtagtag caataataat agcaatagtt gtgtggtcca tagtaatcat agaatatagg
5701 aaaatattaa gacaaagaaa aatagacagg ttaattgata gactaataga aagagcagaa
5761 gacagtggca atgagagtga aggagaaata tcagcacttg tggagatggg ggtggagatg
5821 gggcaccatg ctccttggga tgttgatgat ctgtagtgct acagaaaaat tgtgggtcac

```
5881 agtctattat ggggtacctg tgtggaagga agcaaccacc actctatttt gtgcatcaga
5941 tgctaaagca tatgatacag aggtacataa tgtttgggcc acacatgcct gtgtacccac
6001 agaccccaac ccacaagaag tagtattggt aaatgtgaca gaaaatttta acatgtggaa
6061 aaatgacatg gtagaacaga tgcatgagga tataatcagt ttatgggatc aaagcctaaa
6121 gccatgtgta aaattaaccc cactctgtgt tagtttaaag tgcactgatt tgaagaatga
6181 tactaatacc aatagtagta gcgggagaat gataatggag aaaggagaga taaaaaactg
6241 ctctttcaat atcagcacaa gcataagagg taaggtgcag aaagaatatg catttttta
6301 taaacttgat ataataccaa tagataatga tactaccagc tataagttga caagttgtaa
6361 cacctcagtc attacacagg cctgtccaaa ggtatccttt gagccaattc ccatacatta
6421 ttgtgccccg gctggttttg cgattctaaa atgtaataat aagacgttca atggaacagg
6481 accatgtaca aatgtcagca cagtacaatg tacacatgga attaggccag tagtatcaac
6541 tcaactgctg ttaaatggca gtctagcaga agaagaggta gtaattagat ctgtcaattt
6601 cacggacaat gctaaaacca taatagtaca gctgaacaca tctgtagaaa ttaattgtac
6661 aagacccaac aacaatacaa gaaaaagaat ccgtatccag agaggaccag ggagagcatt
6721 tgttacaata ggaaaaatag gaaatatgag acaagcacat tgtaacatta gtagagcaaa
6781 atggaataac actttaaaac agatagctag caaattaaga gaacaatttg gaaataataa
6841 aacaataatc tttaagcaat cctcaggagg ggacccagaa attgtaacgc acagttttaa
6901 ttgtggaggg gaatttttct actgtaattc aacacaactg tttaatagta cttggtttaa
6961 tagtacttgg agtactgaag ggtcaaataa cactgaagga agtgacacaa tcaccctccc
7021 atgcagaata aaacaaatta taaacatgtg gcagaaagta ggaaaagcaa tgtatgcccc
7081 tcccatcagt ggacaaatta gatgttcatc aaatattaca gggctgctat taacaagaga
7141 tggtggtaat agcaacaatg agtccgagat cttcagacct ggaggaggag atatgaggga
7201 caattggaga agtgaattat ataaatataa agtagtaaaa attgaaccat taggagtagc
7261 acccaccaag gcaaagagaa gagtggtgca gagagaaaaa agagcagtgg gaataggagc
7321 tttgttcctt gggttcttgg gagcagcagg aagcactatg ggcgcagcct caatgacgct
7381 gacggtacag gccagacaat tattgtctgg tatagtgcag cagcagaaca atttgctgag
7441 ggctattgag gcgcaacagc atctgttgca actcacagtc tggggcatca agcagctcca
7501 ggcaagaatc ctggctgtgg aaagatacct aaaggatcaa cagctcctgg ggatttgggg
7561 ttgctctgga aaactcattt gcaccactgc tgtgccttgg aatgctagtt ggagtaataa
7621 atctctggaa cagatttgga atcacacgac ctggatggag tgggacagag aaattaacaa
7681 ttacacaagc ttaatacact ccttaattga agaatcgcaa aaccagcaag aaaagaatga
7741 acaagaatta ttggaattag ataaatgggc aagtttgtgg aattggttta acataacaaa
7801 ttggctgtgg tatataaaat tattcataat gatagtagga ggcttggtag gtttaagaat
7861 agtttttgct gtactttcta tagtgaatag agttaggcag ggatattcac cattatcgtt
7921 tcagacccac ctcccaaccc cgaggggacc cgacaggccc gaaggaatag aagaagaagg
7981 tggagagaga gacagagaca gatccattcg attagtgaac ggatccttgg cacttatctg
8041 ggacgatctg cggagcctgt gcctcttcag ctaccaccgc ttgagagact tactcttgat
8101 tgtaacgagg attgtggaac ttctgggacg caggggtgg gaagccctca aatattggtg
8161 gaatctccta cagtattgga gtcaggaact aaagaatagt gctgttagct tgctcaatgc
8221 cacagccata gcagtagctg aggggacaga tagggttata gaagtagtac aaggagcttg
8281 tagagctatt cgccacatac ctagaagaat aagacagggc ttggaaagga ttttgctata
8341 agatgggtgg caagtggtca aaaagtagtg tgattggatg gcctactgta agggaaagaa
8401 tgagacgagc tgagccagca gcagataggg tgggagcagc atctcgagac ctggaaaaac
8461 atggagcaat cacaagtagc aatacagcag ctaccaatgc tgcttgtgcc tggctagaag
8521 cacaagagga ggaggaggtg ggttttccag tcacacctca ggtaccttta agaccaatga
8581 cttacaaggc agctgtagat cttagccact ttttaaaaga aaaggggga ctggaagggc
8641 taattcactc ccaaagaaga caagatatcc ttgatctgtg gatctaccac acacaaggct
8701 acttccctga ttagcagaac tacacaccag ggccaggggt cagatatcca ctgacctttg
8761 gatggtgcta caagctagta ccagttgagc cagataagat agaagaggcc aataaaggag
8821 agaacaccag cttgttacac cctgtgagcc tgcatgggat ggatgacccg gagagagaag
8881 tgttagagtg gaggtttgac agccgcctag catttcatca cgtggcccga gagctgcatc
8941 cggagtactt caagaactgc tgacatcgag cttgctacaa gggactttcc gctggggact
9001 ttccagggag gcgtggcctg ggcgggactg gggagtggcg agccctcaga tcctgcatat
9061 aagcagctgc tttttgcctg tactgggtct ctctggttag accagatctg agcctgggag
9121 ctctctggct aactagggaa cccactgctt aagcctcaat aaagcttgcc ttgagtgctt
9181 c
```

La duración del análisis fué: 00:00:55

La secuencia de mRNA utilizada para la traducción fué:

GGUCUCUCUGGUUAGACCAGAUCUGAGCCUGGGAGCUCUCUGGCUAACUAGGGAACCCACUGCUUAAGCCUCAAUAAAGCUUGCCUUGAGUGCUUCAAGU
AGUGUGUGCCCGUCUGUUGUGUGACUCUGGUAACUAGAGAUCCCUCAGACCCUUUUAGUCAGUGUGGAAAAUCUCUAGCAGUGGCGCCCGAACAGGGACC
UGAAAGCGAAAGGGAAACCAGAGGAGCUCUCUCGACGCAGGACUCGGCUUGCUGAAGCGCGCACGGCAAGAGGCGAGGGGCGGCGACUGGUGAGUACGCC

AAAAAUUUUGACUAGCGGAGGCUAGAAGGAGAGAGAUGGGUGCGAGAGCGUCAGUAUUAAGCGGGGGAGAAUUAGAUCGAUGGGAAAAAAUUCGGUUAAG
GCCAGGGGGAAAGAAAAAAUAUAAAUUAAAACAUAUAGUAUGGGCAAGCAGGGAGCUAGAACGAUUCGCAGUUAAUCCUGGCCUGUUAGAAACAUCAGAA
GGCUGUAGACAAAUACUGGGACAGCUACAACCAUCCCUUCAGACAGGAUCAGAAGAACUUAGAUCAUUAUAUAAUACAGUAGCAACCCUCUAUUGUGUGC
AUCAAAGGAUAGAGAUAAAAGACACCAAGGAAGCUUUAGACAAGAUAGAGGAAGAGCAAAACAAAAGUAAGAAAAAAGCACAGCAAGCAGCAGCUGACAC
AGGACACAGCAAUCAGGUCAGCCAAAAUUACCCUAUAGUGCAGAACAUCCAGGGGCAAAUGGUACAUCAGGCCAUAUCACCUAGAACUUUAAAUGCAUGG
GUAAAAGUAGUAGAAGAGAAGGCUUUCAGCCCAGAAGUGAUACCCAUGUUUUCAGCAUUAUCAGAAGGAGCCACCCCACAAGAUUUAAACACCAUGCUAA
ACACAGUGGGGGGACAUCAAGCAGCCAUGCAAAUGUUAAAAGAGACCAUCAAUGAGGAAGCUGCAGAAUGGGAUAGAGUGCAUCCAGUGCAUGCAGGGCC
UAUUGCACCAGGCCAGAUGAGAGAACCAAGGGGAAGUGACAUAGCAGGAACUACUAGUACCCUUCAGGAACAAAUAGGAUGGAUGACAAAUAAUCCACCU
AUCCCAGUAGGAGAAAUUUAUAAAAGAUGGAUAAUCCUGGGAUUAAAUAAAAUAGUAAGAAUGUAUAGCCCUACCAGCAUUCUGGACAUAAGACAAGGAC
CAAAGGAACCCUUUAGAGACUAUGUAGACCGGUUCUAUAAAACUCUAAGAGCCGAGCAAGCUUCACAGGAGGUAAAAAAUUGGAUGACAGAAACCUUGUU
GGUCCAAAAUGCGAACCCAGAUUGUAAGACUAUUUUAAAAGCAUUGGGACCAGCGGCUACACUAGAAGAAAUGAUGACAGCAUGUCAGGGAGUAGGAGGA
CCCGGCCAUAAGGCAAGAGUUUUGGCUGAAGCAAUGAGCCAAGUAACAAAUUCAGCUACCAUAAUGAUGCAGAGAGGCAAUUUUAGGAACCAAAGAAAGA
UUGUUAAGUGUUUCAAUUGUGGCAAAGAAGGGCACACAGCCAGAAAUUGCAGGGCCCCUAGGAAAAAGGGCUGUUGGAAAUGUGGAAAGGAAGGACACCA
AAUGAAAGAUUGUACUGAGAGACAGGCUAAUUUUUUAGGGAAGAUCUGGCCUUCCUACAAGGGAAGGCCAGGGAAUUUUCUUCAGAGCAGACCAGAGCCA
ACAGCCCCACCAGAAGAGAGCUUCAGGUCUGGGGUAGAGACAACAACUCCCCCUCAGAAGCAGGAGCCGAUAGACAAGGAACUGUAUCCUUUAACUUCCC
UCAGGUCACUCUUUGGCAACGACCCCUCGUCACAAUAAAGAUAGGGGGGCAACUAAAGGAAGCUCUAUUAGAUACAGGAGCAGAUGAUACAGUAUUAGAA
GAAAUGAGUUUGCCAGGAAGAUGGAAACCAAAAAUGAUAGGGGGAAUUGGAGGUUUUAUCAAAGUAAGACAGUAUGAUCAGAUACUCAUAGAAAUCUGUG
GACAUAAAGCUAUAGGUACAGUAUUAGUAGGACCUACACCUGUCAACAUAAUUGGAAGAAAUCUGUUGACUCAGAUUGGUUGCACUUUAAAUUUUCCCAU
UAGCCCUAUUGAGACUGUACCAGUAAAAUUAAAGCCAGGAAUGGAUGGCCCAAAAGUUAAACAAUGGCCAUUGACAGAAGAAAAAAUAAAAGCAUUAGUA
GAAAUUUGUACAGAGAUGGAAAAGGAAGGGAAAAUUUCAAAAAUUGGGCCUGAAAAUCCAUACAAUACUCCCAGUAUUUGCCAUAAAGAAAAAAGACAGUA
CUAAAUGGAGAAAAUUAGUAGAUUUCAGAGAACUUAAUAAGAGAACUCAAGACUUCUGGGAAGUUCAAUUAGGAAUACCACAUCCCGCAGGGUUAAAAAA
GAAAAAAUCAGUAACAGUACUGGAUGUGGGUGAUGCAUAUUUUUCAGUUCCCUUAGAUGAAGACUUCAGGAAGUAUACUGCAUUUACCAUACCUAGUAUA
AACAAUGAGACACCAGGGAUUAGAUAUCAGUACAAUGUGCUUCCACAGGGAUGGAAAGGAUCACCAGCAAUAUUCCAAAGUAGCAUGACAAAAAUCUUAG
AGCCUUUUAGAAAACAAAAUCCAGACAUAGUUAUCUAUCAAUACAUGGAUGAUUUGUAUGUAGGAUCUGACUUAGAAAUAGGGCAGCAUAGAACAAAAAU
AGAGGAGCUGAGACAACAUCUGUUGAGGUGGGGACUUACCACACCAGACAAAAAACAUCAGAAAGAACCUCCAUUCCUUUGGAUGGGUUAUGAACUCCAU
CCUGAUAAAUGGACAGUACAGCCUAUAGUGCUGCCAGAAAAAGACAGCUGGACUGUCAAUGACAUACAGAAGUUAGUGGGGAAAUUGAAUUGGGCAAGUC
AGAUUUACCCAGGGAUUAAAGUAAGGCAAUUAUGUAAACUCCUUAGAGGAACCAAAGCACUAACAGAAGUAAUACCACUAACAGAAGAAGCAGAGCUAGA
ACUGGCAGAAAACAGAGAGAUUCUAAAAGAACCAGUACAUGGAGUGUAUUAUGACCCAUCAAAAGACUUAAUAGCAGAAAUACAGAAGCAGGGGCAAGGC
CAAUGGACAUAUCAAAUUUAUCAAGAGCCAUUUAAAAAUCUGAAAACAGGAAAAUAUGCAAGAAUGAGGGGUGCCCACACUAAUGAUGUAAAACAAUUAA
CAGAGGCAGUGCAAAAAAUAACCACAGAAAGCAUAGUAAUAUGGGGAAAGACUCCUAAAUUUAAACUGCCCAUACAAAAGGAAACAUGGGAAACAUGGUG
GACAGAGUAUUGGCAAGCCACCUGGAUUCCUGAGUGGGAGUUUGUUAAUACCCCUCCCUUAGUGAAAUUAUGGUACCAGUUAGAGAAAGAACCCAUAGUA
GGAGCAGAAACCUUCUAUGUAGAUGGGGCAGCUAACAGGGAGACUAAAUUAGGAAAAGCAGGAUAUGUUACUAAUAGAGGAAGACAAAAAGUUGUCACCC
UAACUGACACAACAAAUCAGAAGACUGAGUUACAAGCAAUUUAUCUAGCUUUGCAGGAUUCGGGAUUAGAAGUAAACAUAGUAACAGACUCACAAUAUGC
AUUAGGAAUCAUUCAAGCACAACCAGAUCAAAGUGAAUCAGAGUUAGUCAAUCAAAUAAUAGAGCAGUUAAUAAAAAAGGAAAAGGUCUAUCUGGCAUGG
GUACCAGCACACAAAGGAAUUGGAGGAAAUGAACAAGUAGAUAAAUUAGUCAGUGCUGGAAUCAGGAAAGUACUAUUUUUAGAUGGAAUAGAUAAGGCCC
AAGAUGAACAUGAGAAAUAUCACAGUAAUUGGAGAGCAAUGGCUAGUGAUUUUAACCUGCCACCUGUAGUAGCAAAAGAAAUAGUAGCCAGCUGUGAUAA
AUGUCAGCUAAAAGGAGAAGCCAUGCAUGGACAAGUAGACUGUAGUCCAGGAAUAUGGCAACUAGAUUGUACACAUUUAGAAGGAAAAGUUAUCCUGGUA
GCAGUUCAUGUAGCCAGUGGAUAUAUAGAAGCAGAAGUUAUUCCAGCAGAAACAGGGCAGGAAACAGCAUAUUUUCUUUUAAAAUUAGCAGGAAGAUGGC
CAGUAAAAACAAUACAUACUGACAAUGGCAGCAAUUUCACCGGUGCUACGGUUAGGGCCGCCUGUUGGUGGGCGGGAAUCAAGCAGGAAUUUGGAAUUCC
CUACAAUCCCCAAAGUCAAGGAGUAGUAGAAUCUAUGAAUAAAGAAUUAAAGAAAAUUAUAGGACAGGUAAGAGAUCAGGCUGAACAUCUUAAGACAGCA
GUACAAAUGGCAGUAUUCAUCCACAAUUUUAAAAGAAAAGGGGGGAUUGGGGGGUACAGUGCAGGGGAAAGAAUAGUAGACAUAAUAGCAACAGACAUAC
AAACUAAAGAAUUACAAAAACAAAUUACAAAAAUUCAAAAUUUUCGGGUUUAUUACAGGGACAGCAGAAAUCCACUUUGGAAAGGACCAGCAAAGCUCCU
CUGGAAAGGUGAAGGGGCAGUAGUAAUACAAGAUAAUAGUGACAUAAAAGUAGUGCCAAGAAGAAAAGCAAAGAUCAUUAGGGAUUAUGGAAAACAGAUG
GCAGGUGAUGAUUGUGUGGCAAGUAGACAGGAUGAGGAUUAGAACAUGGAAAAGUUUAGUAAAACACCAUAUGUAUGUUUCAGGGAAAGCUAGGGGAUGG
UUUUAUAGACAUCACUAUGAAAGCCCUCAUCCAAGAAUAAGUUCAGAAGUACACAUCCCACUAGGGGAUGCUAGAUUGGUAAUAACAACAUAUUGGGGUC
UGCAUACAGGAGAAAGAGACUGGCAUUUGGGUCAGGGAGUCUCCAUAGAAUGGAGGAAAAAGAGAUAUAGCACACAAGUAGACCCUGAACUAGCAGACCA
ACUAAUUCAUCUGUAUUACUUUGACUGUUUUUCAGACUCUGCUAUAAGAAAGGCCUUAUUAGGACACAUAGUUAGCCCUAGGUGUGAAUAUCAAGCAGGA
CAUAACAAGGUAGGAUCUCUACAAUACUUGGCACUAGCAGCAUUAAUAACACCAAAAAAGAUAAAGCCACCUUUGCCUAGUGUUACGAAACUGACAGAGG
AUAGAUGGAACAAGCCCCAGAAGACCAAGGGCCACAGAGGGAGCCACACAAUGAAUGGACACUAGAGCUUUUAGAGGAGCUUAAGAAUGAAGCUGUUAGA
CAUUUUCCUAGGAUUUGGCUCCAUGGCUAGGGCAACAUAUCUAUGAAACUUAUGGGGAUACUUGGGCAGGAGUGGAAGCCAUAAUAAGAAUUCUGCAAC
AACUGCUGUUUAUCCAUUUUCAGAAUUGGGUGUCGACAUAGCAGAAUAGGCGUUACUCGACAGAGGAGAGCAAGAAAUGGAGCCAGUAGAUCCUAGACUA
GAGCCCUGGAAGCAUCCAGGAAGUCAGCCUAAAACUGCUUGUACCAAUUGCUAUUGUAAAAAGUGUUGCUUUCAUUGCCAAGUUUGUUUCAUAACAAAAG
CCUUAGGCAUCUCCUAUGGCAGGAAGAAGCGGAGACAGCGACGAAGAGCUCAUCAGAACAGUCAGACUCAUCAAGCUUCUCUAUCAAAGCAGUAAGUAGU
ACAUGUAAUGCAACCUAUACCAAUAGUAGCAAUAGUAGCAUUAGUAGUAGCAAUAAUAAUAGCAAUAGUUGUGUGGUCCAUAGUAAUCAUAGAAUAUAGG
AAAAUAUUAAGACAAAGAAAAAUAGACAGGUUAAUUGAUAGACUAAUAGAAAGAGCAGAAGACAGUGGCAAUGAGAGUGAAGGAGAAAUAUCAGCACUUG
UGGAGAUGGGGGUGGAGAUGGGGCACCAUGCUCCUUGGGAUGUUGAUGAUCUGUAGUGCUACAGAAAAAUUGUGGGUCACAGUCUAUUAUGGGGUACCUG
UGUGGAAGGAAGCAACCACCACUCUAUUUUGUGCAUCAGAUGCUAAAGCAUAUGAUACAGAGGUACAUAAUGUUUGGGCCACACAUGCCUGUGUACCCAC
AGACCCCAACCCACAAGAAGUAGUAUUGGUAAAUGUGACAGAAAAUUUUAACAUGUGGAAAAAUGACAUGGUAGAACAGAUGCAUGAGGAUAUAAUCAGU
UUAUGGGAUCAAAGCCUAAAGCCAUGUGUAAAAUUAACCCCACUCUGUGUUAGUUUAAAGUGCACUGAUUUGAAGAAUGAUACUAAUACCAAUAGUAGUA
GCGGGAGAAUGAUAAUGGAGAAAGGAGAGAUAAAAAACUGCUCUUUCAAUAUCAGCACAAGCAUAAGAGGUAAGGUGCAGAAAGAAUAUGCAUUUUUUUA
UAAACUUGAUAUAAUACCAAUAGAUAAUGAUACUACCAGCUAUAAGUUGACAAGUUGUAACACCUCAGUCAUUACACAGGCCUGUCCAAAGGUAUCCUUU
GAGCCAAUUCCCAUACAUUAUUGUGCCCCGGCUGGUUUUGCGAUUCUAAAAUGUAAUAAUAAGACGUUCAAUGGAACAGGACCAUGUACAAAUGUCAGCA
CAGUACAAUGUACACAUGGAAUUAGGCCAGUAGUAUCAACUCAACUGCUGUUAAAUGGCAGUCUAGCAGAAGAAGAGGUAGUAAUUAGAUCUGUCAAUUU
CACGGACAAUGCUAAAACCAUAAUAGUACAGCUGAACACAUCUGUAGAAAUUAAUUGUACAAGACCCAACAACAAUACAAGAAAAAGAAUCCGUAUCCAG
AGAGGACCAGGGAGAGCAUUUGUUACAAUAGGAAAAAUAGGAAAUAUGAGACAAGCACAUUGUAACAUUAGUAGAGCAAAAUGGAAUAACACUUUAAAAC
AGAUAGCUAGCAAAUUAAGAGAACAAUUUGGAAAUAAUAAAACAAUAAUCUUUAAGCAAUCCUCAGGAGGGGACCCAGAAAUUGUAACGCACAGUUUUAA
UUGUGGAGGGGAAUUUUUCUACUGUAAUUCAACACAACUGUUUAAUAGUACUUGGUUUAAUAGUACUUGGAGUACUGAAGGGUCAAAUAACACUGAAGGA
AGUGACACAAUCACCCUCCCAUGCAGAAUAAAACAAAUUAUAAACAUGUGGCAGAAAGUAGGAAAAGCAAUGUAUGCCCCUCCCAUCAGUGGACAAAUUA

GAUGUUCAUCAAAUAUUACAGGGCUGCUAUUAACAAGAGAUGGUGGUAAUAGCAACAAUGAGUCCGAGAUCUUCAGACCUGGAGGAGGAGAUAUGAGGGA
CAAUUGGAGAAGUGAAUUAUAUAAAUAUAAAGUAGUAAAAAUUGAACCAUUAGGAGUAGCACCCACCAAGGCAAAGAGAAGAGUGGUGCAGAGAGAAAAA
AGAGCAGUGGGAAUAGGAGCUUUGUUCCUUGGGUUCUUGGGAGCAGCAGGAAGCACUAUGGGCGCAGCCUCAAUGACGCUGACGGUACAGGCCAGACAAU
UAUUGUCUGGUAUAGUGCAGCAGCAGAACAAUUUGCUGAGGGCUAUUGAGGCGCAACAGCAUCUGUUGCAACUCACAGUCUGGGGCAUCAAGCAGCUCCA
GGCAAGAAUCCUGGCUGUGGAAAGAUACCUAAAGGAUCAACAGCUCCUGGGGAUUUGGGGUUGCUCUGGAAAACUCAUUUGCACCACUGCUGUGCCUUGG
AAUGCUAGUUGGAGUAAUAAAUCUCUGGAACAGAUUUGGAAUCACACGACCUGGAUGGAGUGGGACAGAGAAAUUAACAAUUACACAAGCUUAAUACACU
CCUUAAUUGAAGAAUCGCAAAACCAGCAAGAAAAGAAUGAACAAGAAUUAUUGGAAUUAGAUAAAUGGGCAAGUUUGUGGAAUUGGUUUAACAUAACAAA
UUGGCUGUGGUAUAUAAAAUUAUUCAUAAUGAUAGUAGGAGGCUUGGUAGGUUUAAGAAUAGUUUUGCUGUACUUUCUAUAGUGAAUAGAGUUAGGCAG
GGAUAUUCACCAUUAUCGUUUCAGACCCACCUCCCAACCCGAGGGGACCCGACAGGCCCGAAGGAAUAGAAGAAGAAGGUGGAGAGAGAGACAGAGACA
GAUCCAUUCGAUUAGUGAACGGAUCCUUGGCACUUAUCUGGGACGAUCUGCGGAGCCUGUGCCUCUUCAGCUACCACCGCUUGAGAGACUUACUCUUGAU
UGUAACGAGGAUUGUGGAACUUCUGGGACGCAGGGGGUGGGAAGCCCUCAAAUAUUGGUGGAAUCUCCUACAGUAUUGGAGUCAGGAACUAAAGAAUAGU
GCUGUUAGCUUGCUCAAUGCCACAGCCAUAGCAGUAGCUGAGGGGACAGAUAGGGUUAUAGAAGUAGUACAAGGAGCUUGUAGAGCUAUUCGCCACAUAC
CUAGAAGAAUAAGACAGGGCUUGGAAAGGAUUUUGCUAUAAGAUGGGUGGCAAGUGGUCAAAAAGUAGUGUGAUUGGAUGGCCUACUGUAAGGGAAAGAA
UGAGACGAGCUGAGCCAGCAGCAGAUAGGGUGGGAGCAGCAUCUCGAGACCUGGAAAAACAUGGAGCAAUCACAAGUAGCAAUACAGCAGCUACCAAUGC
UGCUUGUGCCUGGCUAGAAGCACAAGAGGAGGAGGAGGUGGGUUUUCCAGUCACACCUCAGGUACCUUUAAGACCAAUGACUUACAAGGCAGCUGUAGAU
CUUAGCCACUUUUUAAAAGAAAAGGGGGGACUGGAAGGGCUAAUUCACUCCCAAAGAAGACAAGAUAUCCUUGAUCUGUGGAUCUACCACACACAAGGCU
ACUUCCCUGAUUAGCAGAACUACACACCAGGGCCAGGGGUCAGAUAUCCACUGACCUUUGGAUGGUGCUACAAGCUAGUACCAGUUGAGCCAGAUAAGAU
AGAAGAGGCCAAUAAAGGAGAGAACACCAGCUUGUUACACCCUGUGAGCCUGCAUGGGAUGGAUGACCCGGAGAGAGAAGUGUUAGAGUGGAGGUUUGAC
AGCCGCCUAGCAUUUCAUCACGUGGCCCGAGAGCUGCAUCCGGAGUACUUCAAGAACUGCUGACAUCGAGCUUGCUACAAGGGACUUUCCGCUGGGGACU
UUCCAGGGAGGCGUGGCCUGGGCGGGACUGGGGAGUGGCGAGCCCUCAGAUCCUGCAUAUAAGCAGCUGCUUUUUGCCUGUACUGGGUCUCUCUGGUUAG
ACCAGAUCUGAGCCUGGGAGCUCUCUGGCUAACUAGGGAACCCACUGCUUAAGCCUCAAUAAAGCUUGCCUUGAGUGCUUC

Contiene 9181 nucleótidos y su peso molecular es: 2971827.08000018 Da.

Esta compuesta por:
3272 Adeninas (35.64 %).
2042 Uracilos (22.24 %).
2225 Guaninas (24.23 %).
1642 Citosinas (17.88 %).

El porcentaje de AU es: 57.88 % y el de GC es: 42.12 %.

El Código Genético seleccionado para la traducción fué: Standard.

El 1er marco de lectura genera la secuencia:

GLSG*TRSEPGSSLAN*GTHCLSLNKACLECFK*CVPVCCVTLVTRDPSDPFSQCGKSLAVAPEQGPESERETRGALSTQDSAC*SAHGKRRGAATGEYA
KNFD*RRLEGERWVRERQY*AGEN*IDGKKFG*GQGERKNIN*NI*YGQAGS*NDSQLILAC*KHQKAVDKYWDSYNHPFRQDQKNLDHYIIQ*QPSIVC
IKG*R*KTPRKL*TR*RKSKTKVRKKHSKQQLTQDTAIRSAKITL*CRTSRGKWYIRPYHLEL*MHG*K**KRRLSAQK*YPCFQHYQKEPPHKI*TPC*
TQWGDIKQPCKC*KRPSMRKLQNGIECIQCMQGLLHQAR*ENQGEVT*QELLVPFRNK*DG*QIIHLSQ*EKFIKDG*SWD*IK**ECIALPAFWT*DKD
QRNPLETM*TGSIKL*EPSKLHRR*KIG*QKPCWSKMRTQIVRLF*KHWDQRLH*KK**QHVRE*EDPAIRQEFWLKQ*AK*QIQLP**CREAILGTKER
LLSVSIVAKKGTQPEIAGPLGKRAVGNVERKDTK*KIVLRDRLIF*GRSGLPTREGQGIFFRADQSQQPHQKRASGLG*RQQLPLRSRSR*TRNCIL*LP
SGHSLATTPRHNKDRGATKGSSIRYRSR*YSIRRNEFARKMETKNDRGNWRFYQSKTV*SDTHRNLWT*SYRYSISRTYTCQHNWKKSVDSDWLHFKFSH
*PY*DCTSKIKARNGWPKS*TMAIDRRKNKSISRNLYRDGKGRENFKNWA*KSIQYSSICHKEKRQY*MEKISRFQRT**ENSRLLGSSIRNTTSRRVKK
EKISNSTGCG*CIFFSSLR*RLQEVYCIYHT*YKQ*DTRD*ISVQCASTGMERITSNIPK*HDKNLRAF*KTKSRHSYLSIHG*FVCRI*LRNRAA*NKN
RGAETTSVEVGTYHTRQKTSERTSIPLDGL*TPS**MDSTAYSAARKRQLDCQ*HTEVSGEIELGKSDLPRD*SKAIM*TP*RNQSTNRSNTTNRRSRAR
TGRKQRDSKRTSTWSVL*PIKRLNSRNTEAGARPMDISNLSRAI*KSENRKICKNEGCPH**CKTINRGSAKNNHRKHSNMGKDS*I*TAHTKGNMGNMV
DRVLASHLDS*VGVC*YPSLSEIMVPVRERTHSRSRNLLCRWGS*QGD*IRKSRICY**RKTKSCHPN*HNKSED*VTSNLSSFAGFGIRSKHSNRLTIC
IRNHSSTTRSK*IRVSQSNNRAVNKKGKGLSGMGTSTQRNWRK*TSR*ISQCWNQESTIFRWNR*GPR*T*EISQ*LESNG**F*PATCSSKRNSSQL**
MSAKRRSHAWTSRL*SRNMATRLYTFRRKSYPGSSSCSQWIYRSRSYSSRNRAGNSIFSFKISRKMASKNNTY*QWQQFHRCYG*GRLLVGGNQAGIWNS
LQSPKSRSSRIYE*RIKENYRTGKRSG*TS*DSSTNGSIHPQF*KKRGDWGVQCRGKNSRHNSNRHTN*RITKTNYKNSKFSGLLQGQQKSTLERTSKAP
LER*RGSSNTR***HKSSAKKKSKDH*GLWKTDGR**LCGK*TG*GLEHGKV**NTICMFQGKLGDGFIDITMKALIQE*VQKYTSH*GMLDW**QHIGV
CIQEKETGIWVRESP*NGGKRDIAHK*TLN*QTN*FICITLTVFQTLL*ERPY*DT*LALGVNIKQDITR*DLYNTWH*QH**HQKR*SHLCLVLRN*QR
IDGTSPRRPRATEGATQ*MDTRAFRGA*E*SC*TFS*DLAPWLRATYL*NLWGYLGRSGSHNKNSATTAVYPFSELGVDIAE*ALLDRGEQEMEPVDPRL
EPWKHPGSQPKTACTNCYCKKCCFHCQVCFITKALGISYGRKKRRQRRRAHQNSQTHQASLSKQ*VVHVMQPIPIVAIVALVVAIIIAIVVWSIVIIEYR
KILRQRKIDRLIDRLIERAEDSGNESEGEISALVEMGVEMGHHAPWDVDDL*CYRKIVGHSLLWGTCVEGSNHHSILCIRC*SI*YRGT*CLGHTCLCTH
RPQPTRSSIGKCDRKF*HVEK*HGRTDA*GYNQFMGSKPKAMCKINPTLC*FKVH*FEE*Y*YQ***RENDNGERRDKKLLFQYQHKHKR*GAERICIFL
*T*YNTNR**YYQL*VDKL*HLSHYTGLSKGIL*ANSHTLLCPGWFCDSKM***DVQWNRTMYKCQHSTMYTWN*ASSINSTAVKWQSSRRRGSN*ICQF
HGQC*NHNSTAEHICRN*LYKTQQQYKKKNPYPERTRESICYNRKNRKYETSTL*H**SKME*HFKTDS*QIKRTIWK**NNNL*AILRRGPRNCNAQF*
LWRGIFLL*FNTTV**YLV**YLEY*RVK*H*RK*HNHPPMQNKTNYKHVAESRKSNVCPSHQWTN*MFIKYYRAAINKRWW**QQ*VRDLQTWRRRYEG
QLEK*II*I*SSKN*TIRSSTHQGKEKSGAERKKSSGNRSFVPWVLGSSRKHYGRSLNDADGTGQTIIVWYSAAAEQFAEGY*GATASVATHSLGHQAAP
GKNPGCGKIPKGSTAPGDLGLLWKTHLHHCCALEC*LE**ISGTDLESHDLDGVGQRN*QLHKLNTLLN*RIAKPARKE*TRIIGIR*MGKFVELV*HNK
LAVVYKIIHNDSRRLGRFKNSFCCTFYSE*S*AGIFTIIVSDPPPNPEGTRQARRNRRRRWRERQRQIHSISERILGTYLGRSAEPVPLQLPPLERLTLD
CNEDCGTSGTQGVGSPQILVESPTVLESGTKE*CC*LAQCHSHSSS*GDR*GYRSSTRSL*SYSPHT*KNKTGLGKDFAIRWVASGQKVV*LDGLL*GKE

*DELSQQQIGWEQHLETWKNMEQSQVAIQQLPMLLVPG*KHKRRRRWVFQSHLRYL*DQ*LTRQL*ILATF*KKRGDWKG*FTPKEDKISLICGSTTHKA
TSLISRTTHQGQGSDIH*PLDGATS*YQLSQIR*KRPIKERTPACYTL*ACMGWMTRREKC*SGGLTAA*HFITWPESCIRSTSRTADIELATRDFPLGT
FQGGVAWAGLGSGEPSDPAYKQLLFACTGSLWLDQI*AWELSG*LGNPLLKPQ*SLP*VL

Esta secuencia contiene 243 subsecuencia(s):

<eliminado por no presentar relevancias>

La subsecuencia:

ALLDRGEQEMEPVDPRLEPWKHPGSQPKTACTNCYCKKCCFHCQVCFITKALGISYGRKKRRQRRRAHQNSQTHQASLSKQ

Contiene 81 aminoácidos y su peso molecular es: 10842.12 Da.

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 53
El fragmento concordante fué: GISYGR
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: Amidation site (AMIDATION) en la posición: 56
El fragmento concordante fué: YGRK
El patrón de búsqueda fué: x-G-[RK]-[RK]
La expresión regular equivalente es: .G[RK][RK]

Concordancia con los datos publicados en GenBank:

gene **5377..7970**
/gene="tat"
CDS **join(5377..5591,***7925..7970***)**
/gene="tat"
/note="p14; transcriptional activator; viral regulatory protein required for virus replication; transactivates the viral LTR promoter through interactions with cellular transcription factors; associated with pathogenic effects of the virus; the length of Tat varies depending on virus strain or clade"
/codon_start=1
/product="Tat"
/translation="MEPVDPRLEPWKHPGSQPKTACTNCYCKKCCFHCQVCFITKALG ISYGRKKRRQRRRAHQNSQTHQASLSKQ*PTSQPRGDPTGPKE*"

La subsecuencia:

VVHVMQPIPIVAIVALVVAIIIAIVVWSIVIIEYRKILRQRKIDRLIDRLIERAEDSGNESEGEISALVEMGVEMGHHAPWDVDDL

Contiene 86 aminoácidos y su peso molecular es: 11207.86 Da.

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 59
El fragmento concordante fué: NESE
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 57
El fragmento concordante fué: SGNE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 61
El fragmento concordante fué: SEGE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 72
El fragmento concordante fué: GVEMGH
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Concordancias con los datos publicados en GenBank:

gene **5608..5856**
/gene="vpu"
CDS **5608..5856**
/gene="vpu"
/note="p16; viral protein U; viral accessory protein important for virus replication in vivo; promotes degradation of CD4 and down-regulates cell surface expression of MHC class I proteins; helps mediate efficient virus particle release from infected cells; reported to induce apoptosis by suppressing the nuclear factor kappaB-dependent expression of antiapoptotic factors; may attenuate the level of Env precursor(gp160) biosynthesis; Vpu and gp160 are translated from different reading frames of the same bicistronic mRNA"
/codon_start=1
/product="Vpu"
/translation="MQPIPIVAIVALVVAIIIAIVVWSIVIIEYRKILRQRKIDRLID RLIERAEDSGNESEGEISALVEMGVEMGHHAPWDVDDL"

<eliminado por no presentar relevancias>

La subsecuencia:

AGIFTIIVSDPPPNPEGTRQARRNRRRRWRERQRQIHSISERILGTYLGRSAEPVPLQLPPLERLTLDCNEDCGTSGTQGVGSPQILVESPTVL ESGTKE

Contiene 100 aminoácidos y su peso molecular es: 12928.22 Da.

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 40
El fragmento concordante fué: SER
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 38
El fragmento concordante fué: SISE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 92
El fragmento concordante fué: TVLE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 17
El fragmento concordante fué: GTRQAR
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 45
El fragmento concordante fué: GTYLGR
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 74
El fragmento concordante fué: GTSGTQ
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Concordancia con los datos publicados en GenBank:

gene **5516..8199**
/gene="rev"
CDS **join(*5516..5591*,7925..8199)**
/gene="rev"
/note="p19; regulator of expression of virion proteins; prevents splicing of viral RNA; shuttles unspliced viral RNA to the cytoplasm for expression of viral proteins and

incorporation of full length viral genomic RNA into
virions"
/codon_start=1
/product="Rev"
/translation="*MAGRSGDSDEELIRTVRLIKLLYQS*[N]**PPPNPEGTRQARRNRRRR**
WRERQRQIHSISERILGTYLGRSAEPVPLQLPPLERLTLDCNEDCGTSGTQGVGSPQI
LVESPTVLESGTKE"

<eliminado por no presentar relevancias>

El 2do marco de lectura genera la secuencia:

VSLVRPDLSLGALWLTREPTA*ASIKLALSASSSVCPSVV*LW*LEIPQTLLVSVENL*QWRPNRDLKAKGKPEELSRRRTRLAEARTARGEGRRLVSTP
KILTSGG*KERDGCESVSIKRGRIRSMGKNSVKARGKEKI*IKTYSMGKQGARTIRS*SWPVRNIRRL*TNTGTATTIPSDRIRRT*III*YSSNPLLCA
SKDRDKRHQGSFRQDRGRAKQK*EKSTASSS*HRTQQSGQPKLPYSAEHPGANGTSGHIT*NFKCMGKSSRREGFQPRSDTHVFSIIRRSHPTRFKHHAK
HSGGTSSSHANVKRDHQ*GSCRMG*SASSACRAYCTRPDERTKGK*HSRNY*YPSGTNRMDDK*STYPSRRNL*KMDNPGIK*NSKNV*PYQHSGHKTRT
KGTL*RLCRPVL*NSKSRASFTGGKKLDDRNLVGPKCEPRL*DYFKSIGTSGYTRRNDDSMSGSRRTRP*GKSFG*SNEPSNKFSYHNDAERQF*EPKKD
C*VFQLWQRRAHSQKLQGP*EKGLLEMWKGRTPNERLY*ETG*FFREDLAFLQGKAREFSSEQTRANSPTRRELQVWGRDNNSPSEAGADRQGTVSFNFP
QVTLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNFPI
SPIETVPVKLKPGMDGPKVKQWPLTEEKIKALVEICTEMEKEGKISKIGPENPYNTPVFAIKKKDSTKWRKLVDFRELNKRTQDFWEVQLGIPHPAGLKK
KKSVTVLDVGDAYFSVPLDEDFRKYTAFTIPSINNETPGIRYQYNVLPQGWKGSPAIFQSSMTKILEPFRKQNPDIVIYQYMDDLYVGSDLEIGQHRTKI
EELRQHLLRWGLTTPDKKHQKEPPFLWMGYELHPDKWTVQPIVLPEKDSWTVNDIQKLVGKLNWASQIYPGIKVRQLCKLLRGTKALTEVIPLTEEAELE
LAENREILKEPVHGVYYDPSKDLIAEIQKQGQGQWTYQIYQEPFKNLKTGKYARMRGAHTNDVKQLTEAVQKITTESIVIWGKTPKFKLPIQKETWETWW
TEYWQATWIPEWEFVNTPPLVKLWYQLEKEPIVGAETFYVDGAANRETKLGKAGYVTNRGRQKVVTLTDTTNQKTELQAIYLALQDSGLEVNIVTDSQYA
LGIIQAQPDQSESELVNQIIEQLIKKEKVYLAWVPAHKGIGGNEQVDKLVSAGIRKVLFLDGIDKAQDEHEKYHSNWRAMASDFNLPPVVAKEIVASCDK
CQLKGEAMHGQVDCSPGIWQLDCTHLEGKVILVAVHVASGYIEAEVIPAETGQETAYFLLKLAGRWPVKTIHTDNGSNFTGATVRAACWWAGIKQEFGIP
YNPQSQGVVESMNKELKKIIGQVRDQAEHLKTAVQMAVFIHNFKRKGGIGGYSAGERIVDIIATDIQTKELQKQITKIQNFRVYYRDSRNPLWKGPAKLL
WKGEGAVVIQDNSDIKVVPRRKAKIIRDYGKQMAGDDCVASRQDED*NMEKFSKTPYVCFRES*GMVL*TSL*KPSSKNKFRSTHPTRGC*IGNNNILGS
AYRRKRLAFGSGSLHRMEEKEI*HTSRP*TSRPTNSSVLL*LFFRLCYKKGLIRTHS*P*V*ISSRT*QGRISTILGTSSINNTKKDKATFA*CYETDRG
*MEQAPEDQGPQREPHNEWTLELLEELKNEAVRHFPRIWLHGLGQHIYETYGDTWAGVEAIIRILQQLLFIHFQNWVST*QNRRYSTEESKKWSQ*ILD*
SPGSIQEVSLKLLVPIAIVKSVAFIAKFVS*QKP*ASPMAGRSGDSDEELIRTVRLIKLLYQSSK*YM*CNLYQ**Q**H***Q***Q*LCGP**S*NIG
KY*DKEK*TG*LID**KEQKTVAMRVKEKYQHLWRWGWRWGTMLLGMLMICSATEKLWVTVYYGVPVWKEATTTLFCASDAKAYDTEVHNVWATHACVPT
DPNPQEVVLVNVTENFNMWKNDMVEQMHEDIISLWDQSLKPCVKLTPLCVSLKCTDLKNDTNTNSSSGRMIMEKGEIKNCSFNISTSIRGKVQKEYAFFY
KLDIIPIDNDTTSYKLTSCNTSVITQACPKVSFEPIPIHYCAPAGFAILKCNNKTFNGTGPCTNVSTVQCTHGIRPVVSTQLLLNGSLAEEEVVIRSVNF
TDNAKTIIVQLNTSVEINCTRPNNNTRKRIRIQRGPGRAFVTIGKIGNMRQAHCNISRAKWNNTLKQIASKLREQFGNNKTIIFKQSSGGDPEIVTHSFN
CGGEFFYCNSTQLFNSTWFNSTWSTEGSNNTEGSDTITLPCRIKQIINMWQKVGKAMYAPPISGQIRCSSNITGLLLTRDGGNSNNESEIFRPGGGDMRD
NWRSELYKYKVVKIEPLGVAPTKAKRRVVQREKRAVGIGALFLGFLGAAGSTMGAASMTLTVQARQLLSGIVQQQNNLLRAIEAQQHLLQLTVWGIKQLQ
ARILAVERYLKDQQLLGIWGCSGKLICTTAVPWNASWSNKSLEQIWNHTTWMEWDREINNYTSLIHSLIEESQNQQEKNEQELLELDKWASLWNWFNITN
WLWYIKLFIMIVGGLVGLRIVFAVLSIVNRVRQGYSPLSFQTHLPTPRGPDRPEGIEEEGGERDRDRSIRLVNGSLALIWDDLRSLCLFSYHRLRDLLLI
VTRIVELLGRRGWEALKYWWNLLQYWSQELKNSAVSLLNATAIAVAEGTDRVIEVVQGACRAIRHIPRRIRQGLERILL*DGWQVVKK*CDWMAYCKGKN
ETS*ASSR*GGSSISRPGKTWSNHK*QYSSYQCCLCLARSTRGGGGGFSSHTSGTFKTNDLQGSCRS*PLFKRKGGTGRANSLPKKTRYP*SVDLPHTRL
LP*LAELHTRARGQISTDLWMVLQASTS*AR*DRRGQ*RREHQLVTPCEPAWDG*PGERSVRVEV*QPPSISSRGPRAASGVLQELLTSSLLQGTFRWGL
SREAWPGRDWGVASPQILHISSCFLPVLGLSG*TRSEPGSSLAN*GTHCLSLNKACLECF

Esta secuencia contiene 87 subsecuencia(s):

<eliminado por no presentar relevancias>

La subsecuencia:

FFREDLAFLQGKAREFSSEQTRANSPTRRELQVWGRDNNSPSEAGADRQGTVSFNFPQVTLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSL
PGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNFPISPIETVPVKLKPGMDGPKVKQWPLTEEKIKA
LVEICTEMEKEGKISKIGPENPYNTPVFAIKKKDSTKWRKLVDFRELNKRTQDFWEVQLGIPHPAGLKKKKSVTVLDVGDAYFSVPLDEDFRKY
TAFTIPSINNETPGIRYQYNVLPQGWKGSPAIFQSSMTKILEPFRKQNPDIVIYQYMDDLYVGSDLEIGQHRTKIEELRQHLLRWGLTTPDKKH
QKEPPFLWMGYELHPDKWTVQPIVLPEKDSWTVNDIQKLVGKLNWASQIYPGIKVRQLCKLLRGTKALTEVIPLTEEAELELAENREILKEPVH
GVYYDPSKDLIAEIQKQGQGQWTYQIYQEPFKNLKTGKYARMRGAHTNDVKQLTEAVQKITTESIVIWGKTPKFKLPIQKETWETWWTEYWQAT
WIPEWEFVNTPPLVKLWYQLEKEPIVGAETFYVDGAANRETKLGKAGYVTNRGRQKVVTLTDTTNQKTELQAIYLALQDSGLEVNIVTDSQYAL
GIIQAQPDQSESELVNQIIEQLIKKEKVYLAWVPAHKGIGGNEQVDKLVSAGIRKVLFLDGIDKAQDEHEKYHSNWRAMASDFNLPPVVAKEIV
ASCDKCQLKGEAMHGQVDCSPGIWQLDCTHLEGKVILVAVHVASGYIEAEVIPAETGQETAYFLLKLAGRWPVKTIHTDNGSNFTGATVRAACW
WAGIKQEFGIPYNPQSQGVVESMNKELKKIIGQVRDQAEHLKTAVQMAVFIHNFKRKGGIGGYSAGERIVDIIATDIQTKELQKQITKIQNFRV
YYRDSRNPLWKGPAKLLWKGEGAVVIQDNSDIKVVPRRKAKIIRDYGKQMAGDDCVASRQDED

Contiene 1003 aminoácidos y su peso molecular es: 131831.8 Da.

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 832
El fragmento concordante fué: NGSN

```
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: cAMP- and cGMP-dependent protein kinase phosphorylation site (CAMP_PHOSPHO_SITE) en la
             posición: 220
El fragmento concordante fué: KKDS
El patrón de búsqueda fué: [RK](2)-x-[ST]
La expresión regular equivalente es: [RK]{2}.[ST]

Se encontró: cAMP- and cGMP-dependent protein kinase phosphorylation site (CAMP_PHOSPHO_SITE) en la
             posición: 257
El fragmento concordante fué: KKKS
El patrón de búsqueda fué: [RK](2)-x-[ST]
La expresión regular equivalente es: [RK]{2}.[ST]

Se encontró: cAMP- and cGMP-dependent protein kinase phosphorylation site (CAMP_PHOSPHO_SITE) en la
             posición: 280
El fragmento concordante fué: RKYT
El patrón de búsqueda fué: [RK](2)-x-[ST]
La expresión regular equivalente es: [RK]{2}.[ST]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 27
El fragmento concordante fué: TRR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 68
El fragmento concordante fué: TIK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 223
El fragmento concordante fué: STK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 506
El fragmento concordante fué: TGK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 541
El fragmento concordante fué: TPK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 614
El fragmento concordante fué: TNR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 840
El fragmento concordante fué: TVR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 27
El fragmento concordante fué: TRRE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 40
El fragmento concordante fué: SPSE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 82
El fragmento concordante fué: TGAD
```

```
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 87
El fragmento concordante fué: TVLE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 158
El fragmento concordante fué: SPIE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 194
El fragmento concordante fué: TEME
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 262
El fragmento concordante fué: TVLD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 346
El fragmento concordante fué: SDLE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 355
El fragmento concordante fué: TKIE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 370
El fragmento concordante fué: TTPD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 408
El fragmento concordante fué: TVND
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 623
El fragmento concordante fué: TLTD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 644
El fragmento concordante fué: SGLE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 668
El fragmento concordante fué: SESE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 781
El fragmento concordante fué: THLE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 808
El fragmento concordante fué: TGQE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]
```

```
Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 910
El fragmento concordante fué: SAGE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 998
El fragmento concordante fué: SRQD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Tyrosine kinase phosphorylation site (TYR_PHOSPHO_SITE) en la posición: 204
El fragmento concordante fué: KIGPENPY
El patrón de búsqueda fué: [RK]-x(2,3)-[DE]-x(2,3)-Y
La expresión regular equivalente es: [RK].{2,3}[DE].{2,3}Y

Se encontró: Tyrosine kinase phosphorylation site (TYR_PHOSPHO_SITE) en la posición: 328
El fragmento concordante fué: KQNPDIVIY
El patrón de búsqueda fué: [RK]-x(2,3)-[DE]-x(2,3)-Y
La expresión regular equivalente es: [RK].{2,3}[DE].{2,3}Y

Se encontró: Tyrosine kinase phosphorylation site (TYR_PHOSPHO_SITE) en la posición: 722
El fragmento concordante fué: KAQDEHEKY
El patrón de búsqueda fué: [RK]-x(2,3)-[DE]-x(2,3)-Y
La expresión regular equivalente es: [RK].{2,3}[DE].{2,3}Y

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 83
El fragmento concordante fué: GADDTV
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 104
El fragmento concordante fué: GGIGGF
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 150
El fragmento concordante fué: GCTLNF
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 351
El fragmento concordante fué: GQHRTK
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 514
El fragmento concordante fué: GAHTND
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 645
El fragmento concordante fué: GLEVNI
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 659
El fragmento concordante fué: GIIQAQ
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 696
El fragmento concordante fué: GIGGNE
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 719
El fragmento concordante fué: GIDKAQ
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
```

La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 767
El fragmento concordante fué: GQVDCS
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 809
El fragmento concordante fué: GQETAY
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 833
El fragmento concordante fué: GSNFTG
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 864
El fragmento concordante fué: GVVESM
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 904
El fragmento concordante fué: GGIGGY
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: Eukaryotic and viral aspartyl proteases active site (ASP_PROTEASE) en la posición: 78
El fragmento concordante fué: ALLDTGADDTVL
El patrón de búsqueda fué: [LIVMFGAC]-[LIVMTADN]-[LIVFSA]-D-[ST]-G-[STAV]-[STAPDENQ]-x-[LIVMFSTNC]-x-[LIVMFGTA]
La expresión regular equivalente es: [LIVMFGAC][LIVMTADN][LIVFSA]D[ST]G[STAV][STAPDENQ].[LIVMFSTNC].[LIVMFGTA]

Concordancia con los datos publicados en GenBank:

gene **336..4642**
/gene="gag-pol"
CDS **join(***336..1637***,1637..4642)**
/gene="gag-pol"
/note="fusion protein consisting of the viral structural proteins and enzymes; cleaved by the viral protease into individual mature proteins; The processing products of the Gag and Gag-Pol polyproteins were annotated with the help of Pettit et al., 2003 and references therein; Pr160; ribosomal slippage at slippery sequence ttttta (1631..1637)"
/codon_start=1
/product="Gag-Pol"
/translation="*MGARASVLSGGELDRWEKIRLRPGGKKKYKLKHIVWASRELERF*
AVNPGLLETSEGCRQILGQLQPSLQTGSEELRSLYNTVATLYCVHQRIEIKDTKEALD
KIEEEQNKSKKKAQQAAADTGHSNQVSQNYPIVQNIQGQMVHQAISPRTLNAWVKVVE
EKAFSPEVIPMFSALSEGATPQDLNTMLNTVGGHQAAMQMLKETINEEAAEWDRVHPV
HAGPIAPGQMREPRGSDIAGTTSTLQEQIGWMTNNPPIPVGEIYKRWIILGLNKIVRM
YSPTSILDIRQGPKEPFRDYVDRFYKTLRAEQASQEVKNWMTETLLVQNANPDCKTIL
KALGPAATLEEMMTACQGVGGPGHKARVLAEAMSQVTNSATIMMQRGNFRNQRKIVKC
*FNCGKEGHTARNCRAPRKKGCWKCGKEGHQMKDCTERQANFL***REDLAFLQGKAREFSS**
EQTRANSPTRRELQVWGRDNNSPSEAGADRQGTVSFNFPQVTLWQRPLVTIKIGGQLK
EALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVG
PTPVNIIGRNLLTQIGCTLNFPISPIETVPVKLKPGMDGPKVKQWPLTEEKIKALVEI
CTEMEKEGKISKIGPENPYNTPVFAIKKKDSTKWRKLVDFRELNKRTQDFWEVQLGIP
HPAGLKKKKSVTVLDVGDAYFSVPLDEDFRKYTAFTIPSINNETPGIRYQYNVLPQGW
KGSPAIFQSSMTKILEPFRKQNPDIVIYQYMDDLYVGSDLEIGQHRTKIEELRQHLLR
WGLTTPDKKHQKEPPFLWMGYELHPDKWTVQPIVLPEKDSWTVNDIQKLVGKLNWASQ
IYPGIKVRQLCKLLRGTKALTEVIPLTEEAELELAENREILKEPVHGVYYDPSKDLIA
EIQKQGQGQWTYQIYQEPFKNLKTGKYARMRGAHTNDVKQLTEAVQKITTESIVIWGK
TPKFKLPIQKETWETWWTEYWQATWIPEWEFVNTPPLVKLWYQLEKEPIVGAETFYVD
GAANRETKLGKAGYVTNRGRQKVVTLTDTTNQKTELQAIYLALQDSGLEVNIVTDSQY

```
                ALGIIQAQPDQSESELVNQIIEQLIKKEKVYLAWVPAHKGIGGNEQVDKLVSAGIRKV
                LFLDGIDKAQDEHEKYHSNWRAMASDFNLPPVVAKEIVASCDKCQLKGEAMHGQVDCS
                PGIWQLDCTHLEGKVILVAVHVASGYIEAEVIPAETGQETAYFLLKLAGRWPVKTIHT
                DNGSNFTGATVRAACWWAGIKQEFGIPYNPQSQGVVESMNKELKKIIGQVRDQAEHLK
                TAVQMAVFIHNFKRKGGIGGYSAGERIVDIIATDIQTKELQKQITKIQNFRVYYRDSR
                NPLWKGPAKLLWKGEGAVVIQDNSDIKVVPRRKAKIIRDYGKQMAGDDCVASRQDED"
mat_peptide     join(1632..1637,1637..1798)
                /gene="gag-pol"
                /product="Gag-Pol Transframe peptide"
                /note="the Glu-Asp-Leu tripeptide (positions 4-6) is a
                specific inhibitor of the HIV-1 protease. Involved in
                regulation of the protease-mediated polyprotein
                processing; p6*; alternative p6 protein"
mat_peptide     1655..4639
                /gene="gag-pol"
                /product="Pol"
                /note="unprocessed Pol polyprotein; includes part of the
                transframe peptide, protease, reverse transcriptase and
                integrase domains."
mat_peptide     1799..2095
                /gene="gag-pol"
                /product="protease"
                /note="The proteinase domain of Gag-Pol (in the form of
                homodimer) mediates all the cleavages in the polyprotein.
                Cleaves itself from the polyprotein late in particle
                assembly; aspartic peptidase"
mat_peptide     2096..3775
                /gene="gag-pol"
                /product="reverse transcriptase"
                /note="transcribes single stranded viral RNA genome into
                double stranded proviral DNA; HIV-1 reverse transcriptase
                is composed of the p66 subunit (this protein) and the p51
                subunit that lacks the RNAse H domain of the larger
                subunit"
mat_peptide     2096..3415
                /gene="gag-pol"
                /product="reverse transcriptase p51 subunit"
                /note="HIV-1 reverse transcriptase is composed of the p66
                subunit and the p51 subunit (this protein) that lacks the
                RNAse H domain of the larger subunit"
mat_peptide     3776..4639
                /gene="gag-pol"
                /product="integrase"
                /note="mediates integration of the viral DNA into the
                infected cell chromosome"

<eliminado por no presentar relevancias>

    La subsecuencia:

    MEQAPEDQGPQREPHNEWTLELLEELKNEAVRHFPRIWLHGLGQHIYETYGDTWAGVEAIIRILQQLLFIHFQNWVST

    Contiene 78 aminoácidos y su peso molecular es: 10692.74 Da.

        Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 49
        El fragmento concordante fué: TYGD
        El patrón de búsqueda fué: [ST]-x(2)-[DE]
        La expresión regular equivalente es: [ST].{2}[DE]

Concordancia con los datos publicados en GenBank:

gene            5105..5396
                /gene="vpr"
CDS             join(5105..5319,5321..5396)
                /gene="vpr"
                /note="p15; viral protein R; viral accessory protein
                important for virus replication in vivo; involved in the
                nuclear import of the HIV-1 preintegration complex;
```

induces G2 cell cycle arrest; influences mutation rates during viral DNA synthesis; An artificial frameshift eliminating the orf-disrupting nucleotide at position 5320 is introduced to obtain the typical HIV-1 Vpr protein sequence. For this particular HIV-1 strain, HXB2, only a short (78 amino acid long) variant of the Vpr sequence can be obtained by translation of nucleotides 5105 through 5341 without the frameshift"
/codon_start=1
/product="Vpr"
/translation="MEQAPEDQGPQREPHNEWTLELLEELKNEAVRHFPRIWLHGLGQ HIYETYGDTWAGVEAIIRILQQLLFIHF*RIGCRHSRIGVTRQRRARNGASRS*"

<eliminado por no presentar relevancias>

La subsecuencia:

ASPMAGRSGDSDEELIRTVRLIKLLYQSSK

Contiene 30 aminoácidos y su peso molecular es: 3844.28 Da.

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 18
El fragmento concordante fué: TVR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 28
El fragmento concordante fué: SSK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 11
El fragmento concordante fué: SDEE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Concordancia con los datos publicados en GenBank:

gene **5516..8199**
/gene="rev"
CDS **join(5516..5591,***7925..8199***)**
/gene="rev"
/note="p19; regulator of expression of virion proteins; prevents splicing of viral RNA; shuttles unspliced viral RNA to the cytoplasm for expression of viral proteins and incorporation of full length viral genomic RNA into virions"
/codon_start=1
/product="Rev"
/translation="MAGRSGDSDEELIRTVRLIKLLYQS[N]*PPPNPEGTRQARRNRRRR WRERQRQIHSISERILGTYLGRSAEPVPLQLPPLERLTLDCNEDCGTSGTQGVGSPQI LVESPTVLESGTKE*"

<eliminado por no presentar relevancias>

La subsecuencia:

KEQKTVAMRVKEKYQHLWRWGWRWGTMLLGMLMICSATEKLWVTVYYGVPVWKEATTTLFCASDAKAYDTEVHNVWATHACVPTDPNPQEVVLV
NVTENFNMWKNDMVEQMHEDIISLWDQSLKPCVKLTPLCVSLKCTDLKNDTNTNSSSGRMIMEKGEIKNCSFNISTSIRGKVQKEYAFFYKLDI
IPIDNDTTSYKLTSCNTSVITQACPKVSFEPIPIHYCAPAGFAILKCNNKTFNGTGPCTNVSTVQCTHGIRPVVSTQLLLNGSLAEEEVVIRSV
NFTDNAKTIIVQLNTSVEINCTRPNNNTRKRIRIQRGPGRAFVTIGKIGNMRQAHCNISRAKWNNTLKQIASKLREQFGNNKTIIFKQSSGGDP
EIVTHSFNCGGEFFYCNSTQLFNSTWFNSTWSTEGSNNTEGSDTITLPCRIKQIINMWQKVGKAMYAPPISGQIRCSSNITGLLLTRDGGNSNN
ESEIFRPGGGDMRDNWRSELYKYKVVKIEPLGVAPTKAKRRVVQREKRAVGIGALFLGFLGAAGSTMGAASMTLTVQARQLLSGIVQQQNNLLR
AIEAQQHLLQLTVWGIKQLQARILAVERYLKDQQLLGIWGCSGKLICTTAVPWNASWSNKSLEQIWNHTTWMEWDREINNYTSLIHSLIEESQN
QQEKNEQELLELDKWASLWNWFNITNWLWYIKLFIMIVGGLVGLRIVFAVLSIVNRVRQGYSPLSFQTHLPTPRGPDRPEGIEEEGGERDRDRS
IRLVNGSLALIWDDLRSLCLFSYHRLRDLLLIVTRIVELLGRRGWEALKYWWNLLQYWSQELKNSAVSLLNATAIAVAEGTDRVIEVVQGACRA
IRHIPRRIRQGLERILL

Contiene 863 aminoácidos y su peso molecular es: 113527.78 Da.

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 95
El fragmento concordante fué: NVTE
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 143
El fragmento concordante fué: NDTN
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 148
El fragmento concordante fué: NSSS
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 163
El fragmento concordante fué: NCSF
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 167
El fragmento concordante fué: NIST
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 193
El fragmento concordante fué: NDTT
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 204
El fragmento concordante fué: NTSV
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 237
El fragmento concordante fué: NKTF
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 241
El fragmento concordante fué: NGTG
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 248
El fragmento concordante fué: NVST
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 269
El fragmento concordante fué: NGSL
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 283
El fragmento concordante fué: NFTD
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 296
El fragmento concordante fué: NTSV
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 302

El fragmento concordante fué: NCTR
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 308
El fragmento concordante fué: NNTR
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 339
El fragmento concordante fué: NISR
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 346
El fragmento concordante fué: NNTL
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 363
El fragmento concordante fué: NKTI
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 393
El fragmento concordante fué: NSTQ
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 399
El fragmento concordante fué: NSTW
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 404
El fragmento concordante fué: NSTW
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 413
El fragmento concordante fué: NNTE
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 455
El fragmento concordante fué: NITG
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 470
El fragmento concordante fué: NESE
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 618
El fragmento concordante fué: NASW
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 623
El fragmento concordante fué: NKSL
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 631
El fragmento concordante fué: NHTT
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

```
Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 644
El fragmento concordante fué: NYTS
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 681
El fragmento concordante fué: NITN
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 757
El fragmento concordante fué: NGSL
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 823
El fragmento concordante fué: NATA
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 38
El fragmento concordante fué: TEK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 122
El fragmento concordante fué: SLK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 135
El fragmento concordante fué: SLK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 151
El fragmento concordante fué: SGR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 171
El fragmento concordante fué: SIR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 197
El fragmento concordante fué: SYK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 310
El fragmento concordante fué: TRK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 348
El fragmento concordante fué: TLK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 606
El fragmento concordante fué: SGK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 622
El fragmento concordante fué: SNK
```

```
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 730
El fragmento concordante fué: TPR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 752
El fragmento concordante fué: SIR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 833
El fragmento concordante fué: TDR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 36
El fragmento concordante fué: SATE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 117
El fragmento concordante fué: SLWD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 271
El fragmento concordante fué: SLAE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 297
El fragmento concordante fué: TSVE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 372
El fragmento concordante fué: SGGD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 468
El fragmento concordante fué: SNNE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 634
El fragmento concordante fué: TWME
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 651
El fragmento concordante fué: SLIE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Tyrosine kinase phosphorylation site (TYR_PHOSPHO_SITE) en la posición: 794
El fragmento concordante fué: RRGWEALKY
El patrón de búsqueda fué: [RK]-x(2,3)-[DE]-x(2,3)-Y
La expresión regular equivalente es: [RK].{2,3}[DE].{2,3}Y

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 242
El fragmento concordante fué: GTGPCT
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]
```

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 361
El fragmento concordante fué: GNNKTI
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 411
El fragmento concordante fué: GSNNTE
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 448
El fragmento concordante fué: GQIRCS
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 458
El fragmento concordante fué: GLLLTR
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 465
El fragmento concordante fué: GGNSNN
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 502
El fragmento concordante fué: GVAPTK
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 531
El fragmento concordante fué: GAAGST
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 601
El fragmento concordante fué: GIWGCS
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 697
El fragmento concordante fué: GGLVGL
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 842
El fragmento concordante fué: GACRAI
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: Amidation site (AMIDATION) en la posición: 792
El fragmento concordante fué: LGRR
El patrón de búsqueda fué: x-G-[RK]-[RK]
La expresión regular equivalente es: .G[RK][RK]

Se encontró: Leucine zipper pattern (LEUCINE_ZIPPER) en la posición: 800
El fragmento concordante fué: LKYWWNLLQYWSQELKNSAVSL
El patrón de búsqueda fué: L-x(6)-L-x(6)-L-x(6)-L
La expresión regular equivalente es: L.{6}L.{6}L.{6}L

Concordancias con los datos publicados en GenBank:

gene 5771..8341
/gene="env"
CDS 5771..8341
/gene="env"
/note="gp160; envelope glycoprotein; envelope polyprotein;
cleaved by cellular proteases into mature proteins gp120

```
                     and gp41"
                     /codon_start=1
                     /product="Env"
                     /translation="MRVKEKYQHLWRWGWRWGTMLLGMLMICSATEKLWVTVYYGVPV
                     WKEATTTLFCASDAKAYDTEVHNVWATHACVPTDPNPQEVVLVNVTENFNMWKNDMVE
                     QMHEDIISLWDQSLKPCVKLTPLCVSLKCTDLKNDTNTNSSSGRMIMEKGEIKNCSFN
                     ISTSIRGKVQKEYAFFYKLDIIPIDNDTTSYKLTSCNTSVITQACPKVSFEPIPIHYC
                     APAGFAILKCNNKTFNGTGPCTNVSTVQCTHGIRPVVSTQLLLNGSLAEEEVVIRSVN
                     FTDNAKTIIVQLNTSVEINCTRPNNNTRKRIRIQRGPGRAFVTIGKIGNMRQAHCNIS
                     RAKWNNTLKQIASKLREQFGNNKTIIFKQSSGGDPEIVTHSFNCGGEFFYCNSTQLFN
                     STWFNSTWSTEGSNNTEGSDTITLPCRIKQIINMWQKVGKAMYAPPISGQIRCSSNIT
                     GLLLTRDGGNSNNESEIFRPGGGDMRDNWRSELYKYKVVKIEPLGVAPTKAKRRVVQR
                     EKRAVGIGALFLGFLGAAGSTMGAASMTLTVQARQLLSGIVQQQNNLLRAIEAQQHLL
                     QLTVWGIKQLQARILAVERYLKDQQLLGIWGCSGKLICTTAVPWNASWSNKSLEQIWN
                     HTTWMEWDREINNYTSLIHSLIEESQNQQEKNEQELLELDKWASLWNWFNITNWLWYI
                     KLFIMIVGGLVGLRIVFAVLSIVNRVRQGYSPLSFQTHLPTPRGPDRPEGIEEEGGER
                     DRDRSIRLVNGSLALIWDDLRSLCLFSYHRLRDLLLIVTRIVELLGRRGWEALKYWWN
                     LLQYWSQELKNSAVSLLNATAIAVAEGTDRVIEVVQGACRAIRHIPRRIRQGLERILL
                     "
sig_peptide          5771..5854
                     /gene="env"
mat_peptide          5855..7303
                     /gene="env"
                     /product="Envelope surface glycoprotein gp120"
                     /note="mediates binding of HIV-1 to CD4 and cellular
                     co-receptors; cooperates with gp41 to mediate fusion of
                     viral membrane with cellular membrane during virus entry
                     into cells; Envelope surface unit; SU"
mat_peptide          7304..8338
                     /gene="env"
                     /product="Envelope transmembrane glycoprotein gp41"
                     /note="cooperates with gp120 to mediate fusion of viral
                     membrane with cellular membrane during virus entry into
                     cells; Envelope transmembrane domain; TM"
```

<eliminado por no presentar relevancias>

El 3er marco de lectura genera la secuencia:

```
SLWLDQI*AWELSG*LGNPLLKPQ*SLP*VLQVVCARLLCDSGN*RSLRPF*SVWKISSSGARTGT*KRKGNQRSSLDAGLGLLKRARQEARGGDW*VRQ
KF*LAEARRREMGARASVLSGGELDRWEKIRLRPGGKKKYKLKHIVWASRELERFAVNPGLLETSEGCRQILGQLQPSLQTGSEELRSLYNTVATLYCVH
QRIEIKDTKEALDKIEEEQNKSKKKAQQAAADTGHSNQVSQNYPIVQNIQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEGATPQDLNTMLN
TVGGHQAAMQMLKETINEEAAEWDRVHPVHAGPIAPGQMREPRGSDIAGTTSTLQEQIGWMTNNPPIPVGEIYKRWIILGLNKIVRMYSPTSILDIRQGP
KEPFRDYVDRFYKTLRAEQASQEVKNWMTETLLVQNANPDCKTILKALGPAATLEEMMTACQGVGGPGHKARVLAEAMSQVTNSATIMMQRGNFRNQRKI
VKCFNCGKEGHTARNCRAPRKKGCWKCGKEGHQMKDCTERQANFLGKIWPSYKGRPGNFLQSRPEPTAPPEESFRSGVETTTPPQKQEPIDKELYPLTSL
RSLFGNDPSSQ*R*GGN*RKLY*IQEQMIQY*KK*VCQEDGNQK**GELEVLSK*DSMIRYS*KSVDIKL*VQY**DLHLST*LEEIC*LRLVAL*IFPL
ALLRLYQ*N*SQEWMAQKLNNGH*QKKK*KH**KFVQRWKRKGKFQKLGLKIHTILQYLP*RKKTVLNGEN**ISENLIRELKTSGKFN*EYHIPQG*KR
KNQ*QYWMWVMHIFQFP*MKTSGSILHLPYLV*TMRHQGLDISTMCFHRDGKDHQQYSKVA*QKS*SLLENKIQT*LSINTWMICM*DLT*K*GSIEQK*
RS*DNIC*GGDLPHQTKNIRKNLHSFGWVMNSILINGQYSL*CCQKKTAGLSMTYRS*WGN*IGQVRFTQGLK*GNYVNSLEEPKH*QK*YH*QKKQS*N
WQKTERF*KNQYMECIMTHQKT**QKYRSRGKANGHIKFIKSHLKI*KQENMQE*GVPTLMM*NN*QRQCKK*PQKA**YGERLLNLNCPYKRKHGKHGG
QSIGKPPGFLSGSLLIPLP**NYGTS*RKNP**EQKPSM*MGQLTGRLN*EKQDMLLIEEDKKLSP*LTQQIRRLSYKQFI*LCRIRD*K*T**QTHNMH
*ESFKHNQIKVNQS*SIK**SS**KRKRSIWHGYQHTKELEEMNK*IN*SVLESGKYYF*ME*IRPKMNMRNITVIGEQWLVILTCHL**QKK**PAVIN
VS*KEKPCMDK*TVVQEYGN*IVHI*KEKLSW*QFM*PVDI*KQKLFQQKQGRKQHIFF*N*QEDGQ*KQYILTMAAISPVLRLGPPVGGRESSRNLEFP
TIPKVKE**NL*IKN*RKL*DR*EIRLNILRQQYKWQYSSTILKEKGGLGGTVQGKE**T**QQTYKLKNYKNKLQKFKIFGFITGTAEIHFGKDQQSSS
GKVKGQ**YKIIVT*K*CQEEKQRSLGIMENRWQVMIVWQVDRMRIRTWKSLVKHHMYVSGKARGWFYRHHYESPHPRISSEVHIPLGDARLVITTYWGL
HTGERDWHLGQGVSIEWRKKRYSTQVDPELADQLIHLYYFDCFSDSAIRKALLGHIVSPRCEYQAGHNKVGSLQYLALAALITPKKIKPPLPSVTKLTED
RWNKPQKTKGHRGSHTMNGH*SF*RSLRMKLLDIFLGFGSMA*GNISMKLMGILGQEWKP**EFCNNCCLSIFRIGCRHSRIGVTRQRRARNGASRS*TR
ALEASRKSA*NCLYQLLL*KVLLSLPSLFHNKSLRHLLWQEEAETATKSSSEQSDSSSFSIKAVSSTCNATYTNSSNSSISSSNNNSNSCVVHSNHRI*E
NIKTKKNRQVN**TNRKSRRQWQ*E*RRNISTCGDGGGDGAPCSLGC**SVVLQKNCGSQSIMGYLCGRKQPPLYFVHQMLKHMIQRYIMFGPHMPVYPQ
TPTHKK*YW*M*QKILTCGKMTW*NRCMRI*SVYGIKA*SHV*N*PHSVLV*SALI*RMILIPIVVAGE**WRKER*KTALSISAQA*EVRCRKNMHFFI
NLI*YQ*IMILPAIS*QVVTPQSLHRPVQRYPLSQFPYIIVPRLVLRF*NVIIRRSMEQDHVQMSAQYNVHMELGQ*YQLNCC*MAV*QKKR**LDLSIS
RTMLKP**YS*THL*KLIVQDPTTIQEKESVSREDQGEHLLQ*EK*EI*DKHIVTLVEQNGITL*NR*LAN*ENNLEIIKQ*SLSNPQEGTQKL*RTVLI
VEGNFSTVIQHNCLIVLGLIVLGVLKGQITLKEVTQSPSHAE*NKL*TCGRK*EKQCMPLPSVDKLDVHQILQGCY*QEMVVIATMSPRSSDLEEEI*GT
IGEVNYINIK**KLNH*E*HPPRQREEWCREKKEQWE*ELCSLGSWEQQEALWAQPQ*R*RYRPDNYCLV*CSSRTIC*GLLRRNSICCNSQSGASSSSR
QESWLWKDT*RINSSWGFGVALENSFAPLLCLGMLVGVINLWNRFGITRPGWSGTEKLTITQA*YTP*LKNRKTSKKRMNKNYWN*INGQVCGIGLT*QI
GCGI*NYS****EAW*V*E*FLLYFL**IELGRDIHHYRFRPTSQPRGDPTGPKE*KKKVERETETDPFD**TDPWHLSGTICGACASSATTA*ETYS*L
```

*RGLWNFWDAGGGKPSNIGGISYSIGVRN*RIVLLACSMPQP*Q*LRGQIGL*K*YKELVELFATYLEE*DRAWKGFCYKMGGKWSKSSVIGWPTVRERM
RRAEPAADRVGAASRDLEKHGAITSSNTAATNAACAWLEAQEEEEVGFPVTPQVPLRPMTYKAAVDLSHFLKEKGGLEGLIHSQRRQDILDLWIYHTQGY
FPD*QNYTPGPGVRYPLTFGWCYKLVPVEPDKIEEANKGENTSLLHPVSLHGMDDPEREVLEWRFDSRLAFHHVARELHPEYFKNC*HRACYKGLSAGDF
PGRRGLGGTGEWRALRSCI*AAAFCLYWVSLVRPDLSLGALWLTREPTA*ASIKLALSA

Esta secuencia contiene 213 subsecuencia(s):

<eliminado por no presentar relevancias>

La subsecuencia:

LAEARRREMGARASVLSGGELDRWEKIRLRPGGKKKYKLKHIVWASRELERFAVNPGLLETSEGCRQILGQLQPSLQTGSEELRSLYNTVATLY
CVHQRIEIKDTKEALDKIEEEQNKSKKKAQQAAADTGHSNQVSQNYPIVQNIQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEGAT
PQDLNTMLNTVGGHQAAMQMLKETINEEAAEWDRVHPVHAGPIAPGQMREPRGSDIAGTTSTLQEQIGWMTNNPPIPVGEIYKRWIILGLNKIV
RMYSPTSILDIRQGPKEPFRDYVDRFYKTLRAEQASQEVKNWMTETLLVQNANPDCKTILKALGPAATLEEMMTACQGVGGPGHKARVLAEAMS
QVTNSATIMMQRGNFRNQRKIVKCFNCGKEGHTARNCRAPRKKGCWKCGKEGHQMKDCTERQANFLGKIWPSYKGRPGNFLQSRPEPTAPPEES
FRSGVETTTPPQKQEPIDKELYPLTSLRSLFGNDPSSQ

Contiene 508 aminoácidos y su peso molecular es: 66046.2900000001 Da.

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 117
El fragmento concordante fué: NKSK
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 119
El fragmento concordante fué: SKK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 156
El fragmento concordante fué: SPR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 311
El fragmento concordante fué: TLR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 409
El fragmento concordante fué: TAR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 435
El fragmento concordante fué: TER
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 448
El fragmento concordante fué: SYK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 470
El fragmento concordante fué: SFR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 496
El fragmento concordante fué: SLR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 17
El fragmento concordante fué: SGGE
El patrón de búsqueda fué: [ST]-x(2)-[DE]

```
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 78
El fragmento concordante fué: TGSE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 188
El fragmento concordante fué: TPQD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 212
El fragmento concordante fué: TINE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 250
El fragmento concordante fué: TLQE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 289
El fragmento concordante fué: SILD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 350
El fragmento concordante fué: TLEE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 459
El fragmento concordante fué: SRPE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 473
El fragmento concordante fué: SGVE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: Tyrosine kinase phosphorylation site (TYR_PHOSPHO_SITE) en la posición: 302
El fragmento concordante fué: RDYVDRFY
El patrón de búsqueda fué: [RK]-x(2,3)-[DE]-x(2,3)-Y
La expresión regular equivalente es: [RK].{2,3}[DE].{2,3}Y

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 10
El fragmento concordante fué: GARASV
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 57
El fragmento concordante fué: GLLETS
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 200
El fragmento concordante fué: GGHQAA
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 241
El fragmento concordante fué: GSDIAG
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 389
```

El fragmento concordante fué: GNFRNQ
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 420
El fragmento concordante fué: GCWKCG
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 474
El fragmento concordante fué: GVETTT
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 502
El fragmento concordante fué: GNDPSS
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: Amidation site (AMIDATION) en la posición: 32
El fragmento concordante fué: GGKK
El patrón de búsqueda fué: x-G-[RK]-[RK]
La expresión regular equivalente es: .G[RK][RK]

Concordancias con los datos publicados en GenBank:

gene **336..1838**
/gene="gag"
CDS **336..1838**
/gene="gag"
/note="Pr55; The processing products of the Gag and Gag-Pol polyproteins were annotated with the help of Pettit et al., 2003 and references therein"
/codon_start=1
/product="Gag"
/translation="MGARASVLSGGELDRWEKIRLRPGGKKKYKLKHIVWASRELERF AVNPGLLETSEGCRQILGQLQPSLQTGSEELRSLYNTVATLYCVHQRIEIKDTKEALD KIEEEQNKSKKKAQQAAADTGHSNQVSQNYPIVQNIQGQMVHQAISPRTLNAWVKVVE EKAFSPEVIPMFSALSEGATPQDLNTMLNTVGGHQAAMQMLKETINEEAAEWDRVHPV HAGPIAPGQMREPRGSDIAGTTSTLQEQIGWMTNNPPIPVGEIYKRWIILGLNKIVRM YSPTSILDIRQGPKEPFRDYVDRFYKTLRAEQASQEVKNWMTETLLVQNANPDCKTIL KALGPAATLEEMMTACQGVGGPGHKARVLAEAMSQVTNSATIMMQRGNFRNQRKIVKC FNCGKEGHTARNCRAPRKKGCWKCGKEGHQMKDCTERQANFLGKIWPSYKGRPGNFLQ SRPEPTAPPEESFRSGVETTTPPQKQEPIDKELYPLTSLRSLFGNDPSSQ"
mat_peptide **336..731**
/gene="gag"
/product="matrix"
/note="viral structural protein; forms the outer structural shell of HIV-1 virions; involved in the nuclear import of the HIV-1 preintegration complex; p17"
mat_peptide **732..1424**
/gene="gag"
/product="capsid"
/note="viral structural protein; forms the core of HIV-1 virions; p24"
mat_peptide **1425..1466**
/gene="gag"
/product="p2"
/note="Processing of Gag-Pol by the protease domain dimer starts with cleavage between the p2 and nucleocapsid proteins."
mat_peptide **1467..1631**
/gene="gag"
/product="nucleocapsid"
/note="viral structural protein; coats the genomic RNA inside the virion core; binds and delivers full-length viral RNAs into assembling HIV-1 virions; p7"
mat_peptide **1632..1679**

```
                /gene="gag"
                /product="p1"
                /note="important for virus infectivity, protein
                processing, and genomic RNA dimer stability"
mat_peptide     1680..1835
                /gene="gag"
                /product="p6"
                /note="important for incorporation of Vpr into assembling
                HIV-1 virions; helps mediate efficient virus particle
                release from infected cells"

gene            336..4642
                /gene="gag-pol"
CDS             join(336..1637,1637..4642)
                /gene="gag-pol"
                /note="fusion protein consisting of the viral structural
                proteins and enzymes; cleaved by the viral protease into
                individual mature proteins; The processing products of the
                Gag and Gag-Pol polyproteins were annotated with the help
                of Pettit et al., 2003 and references therein; Pr160;
                ribosomal slippage at slippery sequence ttttta
                (1631..1637)"
                /codon_start=1
                /product="Gag-Pol"
                /translation="MGARASVLSGGELDRWEKIRLRPGGKKKYKLKHIVWASRELERF
                AVNPGLLETSEGCRQILGQLQPSLQTGSEELRSLYNTVATLYCVHQRIEIKDTKEALD
                KIEEEQNKSKKKAQQAAADTGHSNQVSQNYPIVQNIQGQMVHQAISPRTLNAWVKVVE
                EKAFSPEVIPMFSALSEGATPQDLNTMLNTVGGHQAAMQMLKETINEEAAEWDRVHPV
                HAGPIAPGQMREPRGSDIAGTTSTLQEQIGWMTNNPPIPVGEIYKRWIILGLNKIVRM
                YSPTSILDIRQGPKEPFRDYVDRFYKTLRAEQASQEVKNWMTETLLVQNANPDCKTIL
                KALGPAATLEEMMTACQGVGGPGHKARVLAEAMSQVTNSATIMMQRGNFRNQRKIVKC
                FNCGKEGHTARNCRAPRKKGCWKCGKEGHQMKDCTERQANFL
                REDLAFLQGKAREFSS
                EQTRANSPTRRELQVWGRDNNSPSEAGADRQGTVSFNFPQVTLWQRPLVTIKIGGQLK
                EALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVG
                PTPVNIIGRNLLTQIGCTLNFPISPIETVPVKLKPGMDGPKVKQWPLTEEKIKALVEI
                CTEMEKEGKISKIGPENPYNTPVFAIKKKDSTKWRKLVDFRELNKRTQDFWEVQLGIP
                HPAGLKKKKSVTVLDVGDAYFSVPLDEDFRKYTAFTIPSINNETPGIRYQYNVLPQGW
                KGSPAIFQSSMTKILEPFRKQNPDIVIYQYMDDLYVGSDLEIGQHRTKIEELRQHLLR
                WGLTTPDKKHQKEPPFLWMGYELHPDKWTVQPIVLPEKDSWTVNDIQKLVGKLNWASQ
                IYPGIKVRQLCKLLRGTKALTEVIPLTEEAELELAENREILKEPVHGVYYDPSKDLIA
                EIQKQGQGQWTYQIYQEPFKNLKTGKYARMRGAHTNDVKQLTEAVQKITTESIVIWGK
                TPKFKLPIQKETWETWWTEYWQATWIPEWEFVNTPPLVKLWYQLEKEPIVGAETFYVD
                GAANRETKLGKAGYVTNRGRQKVVTLTDTTNQKTELQAIYLALQDSGLEVNIVTDSQY
                ALGIIQAQPDQSESELVNQIIEQLIKKEKVYLAWVPAHKGIGGNEQVDKLVSAGIRKV
                LFLDGIDKAQDEHEKYHSNWRAMASDFNLPPVVAKEIVASCDKCQLKGEAMHGQVDCS
                PGIWQLDCTHLEGKVILVAVHVASGYIEAEVIPAETGQETAYFLLKLAGRWPVKTIHT
                DNGSNFTGATVRAACWWAGIKQEFGIPYNPQSQGVVESMNKELKKIIGQVRDQAEHLK
                TAVQMAVFIHNFKRKGGIGGYSAGERIVDIIATDIQTKELQKQITKIQNFRVYYRDSR
                NPLWKGPAKLLWKGEGAVVIQDNSDIKVVPRRKAKIIRDYGKQMAGDDCVASRQDED"
mat_peptide     join(1632..1637,1637..1798)
                /gene="gag-pol"
                /product="Gag-Pol Transframe peptide"
                /note="the Glu-Asp-Leu tripeptide (positions 4-6) is a
                specific inhibitor of the HIV-1 protease. Involved in
                regulation of the protease-mediated polyprotein
                processing; p6*; alternative p6 protein"
mat_peptide     1655..4639
                /gene="gag-pol"
                /product="Pol"
                /note="unprocessed Pol polyprotein; includes part of the
                transframe peptide, protease, reverse transcriptase and
                integrase domains."
mat_peptide     1799..2095
                /gene="gag-pol"
                /product="protease"
                /note="The proteinase domain of Gag-Pol (in the form of
                homodimer) mediates all the cleavages in the polyprotein.
                Cleaves itself from the polyprotein late in particle
```

```
                assembly; aspartic peptidase"
mat_peptide     2096..3775
                /gene="gag-pol"
                /product="reverse transcriptase"
                /note="transcribes single stranded viral RNA genome into
                double stranded proviral DNA; HIV-1 reverse transcriptase
                is composed of the p66 subunit (this protein) and the p51
                subunit that lacks the RNAse H domain of the larger
                subunit"
mat_peptide     2096..3415
                /gene="gag-pol"
                /product="reverse transcriptase p51 subunit"
                /note="HIV-1 reverse transcriptase is composed of the p66
                subunit and the p51 subunit (this protein) that lacks the
                RNAse H domain of the larger subunit"
mat_peptide     3776..4639
                /gene="gag-pol"
                /product="integrase"
                /note="mediates integration of the viral DNA into the
                infected cell chromosome"

<eliminado por no presentar relevancias>
```

La subsecuencia:

CQEEKQRSLGIMENRWQVMIVWQVDRMRIRTWKSLVKHHMYVSGKARGWFYRHHYESPHPRISSEVHIPLGDARLVITTYWGLHTGERDWHLGQ
GVSIEWRKKRYSTQVDPELADQLIHLYYFDCFSDSAIRKALLGHIVSPRCEYQAGHNKVGSLQYLALAALITPKKIKPPLPSVTKLTEDRWNKP
QKTKGHRGSHTMNGH

Contiene 203 aminoácidos y su peso molecular es: 27424.69 Da.

Se encontró: cAMP- and cGMP-dependent protein kinase phosphorylation site (CAMP_PHOSPHO_SITE) en la posición: 103
El fragmento concordante fué: KRYS
El patrón de búsqueda fué: [RK](2)-x-[ST]
La expresión regular equivalente es: [RK]{2}.[ST]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 31
El fragmento concordante fué: TWK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 43
El fragmento concordante fué: SGK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 141
El fragmento concordante fué: SPR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 166
El fragmento concordante fué: TPK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 107
El fragmento concordante fué: TQVD
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 10
El fragmento concordante fué: GIMENR
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 82

El fragmento concordante fué: GLHTGE
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 93
El fragmento concordante fué: GQGVSI
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Concordancias con los datos publicados en GenBank:

gene 4587..5165
/gene="vif"
CDS 4587..5165
/gene="vif"
/note="p23; viral infectivity factor; viral accessory protein important for virus replication in vivo"
/codon_start=1
/product="Vif"
/translation="MENRWQVMIVWQVDRMRIRTWKSLVKHHMYVSGKARGWFYRHHY ESPHPRISSEVHIPLGDARLVITTYWGLHTGERDWHLGQGVSIEWRKKRYSTQVDPEL ADQLIHLYYFDCFSDSAIRKALLGHIVSPRCEYQAGHNKVGSLQYLALAALITPKKIK PPLPSVTKLTEDRWNKPQKTKGHRGSHTMNGH"

<eliminado por no presentar relevancias>

La subsecuencia:

EFCNNCCLSIFRIGCRHSRIGVTRQRRARNGASRS

Contiene 35 aminoácidos y su peso molecular es: 4638.2 Da.

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 14
El fragmento concordante fué: GCRHSR
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Concordancia con los datos publicados en GenBank:

gene 5105..5396
/gene="vpr"
CDS join(*5105..5319***,5321..5396)**
/gene="vpr"
/note="p15; viral protein R; viral accessory protein important for virus replication in vivo; involved in the nuclear import of the HIV-1 preintegration complex; induces G2 cell cycle arrest; influences mutation rates during viral DNA synthesis; An artificial frameshift eliminating the orf-disrupting nucleotide at position 5320 is introduced to obtain the typical HIV-1 Vpr protein sequence. For this particular HIV-1 strain, HXB2, only a short (78 amino acid long) variant of the Vpr sequence can be obtained by translation of nucleotides 5105 through 5341 without the frameshift"
/codon_start=1
/product="Vpr"
/translation="*MEQAPEDQGPQREPHNEWTLELLEELKNEAVRHFPRIWLHGLGQ HIYETYGDTWAGVEAIIRILQQLLFIHF***RIGCRHSRIGVTRQRRARNGASRS"**

<eliminado por no presentar relevancias>

La subsecuencia:

IELGRDIHHYRFRPTSQPRGDPTGPKE

Contiene 27 aminoácidos y su peso molecular es: 3628.94 Da.

Se encontró: Cell attachment sequence (RGD) en la posición: 19

El fragmento concordante fué: RGD
El patrón de búsqueda fué: R-G-D
La expresión regular equivalente es: RGD

Concordancia con los datos publicados en GenBank:

gene **5377..7970**
/gene="tat"
CDS **join(*5377..5591*,7925..7970)**
/gene="tat"
/note="p14; transcriptional activator; viral regulatory protein required for virus replication; transactivates the viral LTR promoter through interactions with cellular transcription factors; associated with pathogenic effects of the virus; the length of Tat varies depending on virus strain or clade"
/codon_start=1
/product="Tat"
/translation="*MEPVDPRLEPWKHPGSQPKTACTNCYCKKCCFHCQVCFITKALG ISYGRKKRRQRRRAHQNSQTHQASLSKQ***PTSQPRGDPTGPKE"**

<eliminado por no presentar relevancias>

La subsecuencia:

DRAWKGFCYKMGGKWSKSSVIGWPTVRERMRRAEPAADRVGAASRDLEKHGAITSSNTAATNAACAWLEAQEEEEVGFPVTPQVPLRPMTYKAA
VDLSHFLKEKGGLEGLIHSQRRQDILDLWIYHTQGYFPD

Contiene 133 aminoácidos y su peso molecular es: 17325.99 Da.

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 25
El fragmento concordante fué: TVR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 90
El fragmento concordante fué: TYK
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Protein kinase C phosphorylation site (PKC_PHOSPHO_SITE) en la posición: 113
El fragmento concordante fué: SQR
El patrón de búsqueda fué: [ST]-x-[RK]
La expresión regular equivalente es: [ST].[RK]

Se encontró: Casein kinase II phosphorylation site (CK2_PHOSPHO_SITE) en la posición: 25
El fragmento concordante fué: TVRE
El patrón de búsqueda fué: [ST]-x(2)-[DE]
La expresión regular equivalente es: [ST].{2}[DE]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 12
El fragmento concordante fué: GGKWSK
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 51
El fragmento concordante fué: GAITSS
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Se encontró: N-myristoylation site (MYRISTYL) en la posición: 105
El fragmento concordante fué: GGLEGL
El patrón de búsqueda fué: G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}
La expresión regular equivalente es: G[^EDRKHPFYW].{2}[STAGCN][^P]

Concordancia con los datos publicados en GenBank:

gene **8343..8963**

/gene="nef"
CDS **8343..8963**
/gene="nef"
/note="p27; negative factor; viral accessory protein; important for virus replication in vivo; determinant of HIV-1 pathogenesis; down-regulates cell surface CD4 and MHC class I molecules; enhances virus infectivity through interactions with multiple cellular signaling proteins; This particular nucleotide sequence has a premature stop codon in place of a well-conserved tryptophan codon at position 8712-8714 that truncates the HIV1 Nef protein sequence to a 123 amino acids-long N-terminal portion (not shown)"
/codon_start=1
/transl_except=(pos:8712..8714,aa:Trp)
/product="Nef"
/translation="MGGKWSKSSVIGWPTVRERMRRAEPAADRVGAASRDLEKHGAIT SSNTAATNAACAWLEAQEEEEVGFPVTPQVPLRPMTYKAAVDLSHFLKEKGGLEGLIH SQRRQDILDLWIYHTQGYFPD[W]*QNYTPGPGVRYPLTFGWCYKLVPVEPDKIEEANKGE NTSLLHPVSLHGMDDPEREVLEWRFDSRLAFHHVARELHPEYFKNC*"

La subsecuencia:

QNYTPGPGVRYPLTFGWCYKLVPVEPDKIEEANKGENTSLLHPVSLHGMDDPEREVLEWRFDSRLAFHHVARELHPEYFKNC

Contiene 82 aminoácidos y su peso molecular es: 11068.1 Da.

Se encontró: N-glycosylation site (ASN_GLYCOSYLATION) en la posición: 37
El fragmento concordante fué: NTSL
El patrón de búsqueda fué: N-{P}-[ST]-{P}
La expresión regular equivalente es: N[^P][ST][^P]

Concordancia con los datos publicados en GenBank:

gene **8343..8963**
/gene="nef"
CDS **8343..8963**
/gene="nef"
/note="p27; negative factor; viral accessory protein; important for virus replication in vivo; determinant of HIV-1 pathogenesis; down-regulates cell surface CD4 and MHC class I molecules; enhances virus infectivity through interactions with multiple cellular signaling proteins; This particular nucleotide sequence has a premature stop codon in place of a well-conserved tryptophan codon at position 8712-8714 that truncates the HIV1 Nef protein sequence to a 123 amino acids-long N-terminal portion (not shown)"
/codon_start=1
/transl_except=(pos:8712..8714,aa:Trp)
/product="Nef"
/translation="*MGGKWSKSSVIGWPTVRERMRRAEPAADRVGAASRDLEKHGAIT SSNTAATNAACAWLEAQEEEEVGFPVTPQVPLRPMTYKAAVDLSHFLKEKGGLEGLIH SQRRQDILDLWIYHTQGYFPD*[W]**QNYTPGPGVRYPLTFGWCYKLVPVEPDKIEEANKGE NTSLLHPVSLHGMDDPEREVLEWRFDSRLAFHHVARELHPEYFKNC"**

<eliminado por no presentar relevancias>

El 4to marco de lectura genera la secuencia:

EALKASFIEA*AVGSLVSQRAPRLRSGLTRETQYRQKAAAYMQDLRARHSPVPPRPRLPGKSPAESPL*QARCQQFLKYSGCSSRAT**NARRLSNLHSN
TSLSGSSIPCRLTGCNKLVFSPLLASSILSGSTGTSL*HHPKVSGYLTPGPGV*FC*SGK*PCVW*IHRSRISCLLWE*ISPSSPPFSFKKWLRSTAAL*
VIGLKGT*GVTGKPTSSSSCASSQAQAALVAAVLLLVIAPCFSRSRDAAPTLSAAGSARLILSLTVGHPITLLFDHLPPIL*QNPFQALSYSSRYVANSS
TSSLYYFYNPICPLSYCYGCGIEQANSTIL*FLTPIL*EIPPIFEGFPPPASQKFHNPRYNQE*VSQAVVAEEAQAPQIVPDKCQGSVH*SNGSVSVSLS
TFFFYSFGPVGSPRGWEVGLKR*W*ISLPNSIHYRKYSKNYS*TYQASYYHYE*FYIPQPICYVKPIPQTCPFI*FQ*FLFILFLLVLRFFN*GVY*ACV
IVNFSVPLHPGRVIPNLFQRFITPTSIPRHSSGANEFSRATPNPQELLIL*VSFHSQDSCLELLDAPDCELQQMLLRLNSPQQIVLLLHYTRQ*LSGLYR

QRH*GCAHSASCCSQEPKEQSSYSHCSFFSLHHSSLCLGGCYS*WFNFYYFIFI*FTSPIVPHISSSRSEDLGLIVAITTISC**QPCNI**TSNLSTDG
RGIHCFSYFLPHVYNLFYSAWEGDCVTSFSVI*PFSTPSTIKPSTIKQLC*ITVEKFPSTIKTVRYNFWVPS*GLLKDYCFIISKLFS*FASYLF*SVIP
FCSTNVTMCLSHISYFSYCNKCSPWSSLDTDSFSCIVVGSCTINFYRCVQLYYYGFSIVREIDRSNYYLFFC*TAI*QQLS*YYWPNSMCTLYCADICTW
SCSIERLIITF*NRKTSRGTIMYGNWLKGYLWTGLCND*GVTTCQLIAGSIIIYWYYIKFIKKCIFFLHLTSYACADIERAVFYLSFLHYHSPATTIGIS
IILQISAL*TNTEWG*FYTWL*ALIP*TDYILMHLFYHVIFPHVKIFCHIYQYYFLWVGVCGYTGMCGPNIMYLCIICFSI*CTK*SGGCFLPHRYPIID
CDPQFFCSTTDHQHPKEHGAPSPPPSPQVLIFLLHSHCHCLLLFLLVYQLTCLFFFVLIFSYIL*LLWTTQLLLLLLLLLMLLLLLV*VALHVLLTALI
EKLDESDCSDELFVAVSASSCHRRCLRLLL*NKLGNESNTFYNSNWYKQF*ADFLDASRALV*DLLAPFLALLCRVTPILLCRHPILKMDKQQLLQNSYY
GFHSCPSIPISFIDMLP*AMEPNPRKMSNSFILKLL*KL*CPFIVWLPLWPLVFWGLFHLSSVSFVTLGKGGFIFFGVINAASAKYCRDPTLLCPA*YSH
LGLTMCPNKAFLIAESEKQSK*YR*ISWSASSGSTCVLYLFFLHSMETP*PKCQSLSPVCRPQYVVITNLASPSGMCTSELILG*GLS**CL*NHPLAFP
ETYIWCFTKLFHVLILILSTCHTIITCHLFSIIPNDLCFSSWHYFYVTIILYYYCPFTFPEELCWSFPKWISAVPVINPKILNFCNLFL*FFSLYVCCYY
VYYSFPCTVPPNPPFSFKIVDEYCHLYCCLKMFSLISYLSYNFL*FFIHRFYYSLTLGIVGNSKFLLDSRPPTGGPNRSTGEIAAIVSMYCFYWPSSC*F
*KKICCFLPCFCWNNFCFYISTGYMNCYQDNFSF*MCTI*LPYSWTTVYLSMHGFSF*LTFITAGYYFFCYYRWQVKITSHCSPITVIFLMFILGLIYSI
*K*YFPDSSTD*FIYLFISSNSFVCWYPCQIDLFLFY*LLYYLID*L*FTLIWLCLNDS*CIL*VCYYVYF*SRILQS*INCL*LSLLICCVS*GDNFLS
SSISNISCFS*FSLPVSCPIYIEGFCSYYGFFL*LVP*FH*GRGINKLPLRNPGGLPILCPPCFPCFLLYGQFKFRSLSPYYYAFCGYFLHCLC*LFYII
SVGTPHSCIFSCFQIFKWLLINLICPLALPLLLYFCY*VF*WVIIHSMYWFF*NLSVFCQF*LCFFC*WYYFC*CFGSSKEFT*LPYFNPWVNLTCPIQF
PH*LLYVIDSPAVFFWQHYRLYCPFIRMEFITHPKEWRFFLMFFVWCGKSPPQQMLSQLLYFCSMLPYF*VRSYIQIIHVLIDNYVWILFSKRL*DFCHA
TLEYCW*SFPSLWKHIVLISNPWCLIVYTRYGKCSILPEVFI*GN*KICITHIQYCY*FFLF*PCGMWYS*LNFPEVLSSLIKFSEIY*FSPFSTVFFLY
GKYWSIVWIFRPNF*NFPFLFHLCTNFY*CFYFFFCQWPLFNFWAIHSWL*FYWYSLNRANGKI*SATNLSQQISSNYVDRCRSY*YCTYSFMSTDFYEY
LIILSYFDKTSNSPYHFWFPSSWQTHFF*YCIICSCI**SFL*LPPYLYCDEGSLPKSDLREVKGYSSLSIGSCF*GGVVVSTPDLKLSSGGAVGSGLL*
RKFPGLPL*EGQIFPKKLACLSVQSFIWCPSFPHFQQPFFLGALQFLAVCPSLPQLKHLTIFLWFLKLPLCIIMVAEFVTWLIASAKTLALWPGPPTP*H
AVIISSSVAAGPNAFKIVLQSGFAFWTNKVSVIQFFTSCEACSALRVL*NRST*SLKGSFGPCLMSRMLVGLYILTILFNPRIIHLL*ISPTGIGGLFVI
HPICS*RVLVVPAMSLPLGSLIWPGAIGPACTGCTLSHSAASSLMVSFNICMAA*CPPTVFSMVFKSCGVAPSDNAENMGITSGLKAFSSTTFTHAFKVL
GDMA*CTICPWMFCTIG*FWLT*LLCPVSAAACCAFFLLLFCSSSILSKASLVSFISIL*CTQ*RVATVLYNDLSSSDPV*RDGCSCPSICLQPSDVSNR
PGLTANRSSSLLAHTICFNLYFFFPPGLNRIFSHRSNSPPLNTDALAPISLLLASASQNFWRTHQSPPLASCRARFSKPSPASRELLWFPFRFQVPVRAP
LLEIFHTD*KGLRDL*LPESHNRRAHTT*STQGKLY*GLSSGFPS*PESSQAQIWSNQRD

Esta secuencia contiene 137 subsecuencia(s):

<eliminado por no presentar relevancias>

El 5to marco de lectura genera la secuencia:

KHSRQALLRLKQWVP*LARELPGSDLV*PERPSTGKKQLLICRI*GLATPQSRPGHASLESPQRKVPCSKLDVSSS*STPDAALGPRDEMLGGCQTSTLT
LLSPGHPSHAGSQGVTSWCSLLYWPLLSYLAQLVLACSTIQRSVDI*PLALVCSSANQGSSLVCGRSTDQGYLVFFGSELALPVPPFLLKSG*DLQLPCK
SLVLKVPEV*LENPPPPPLVLLARHKQHW*LLYCYL*LLHVFPGLEMLLPPYLLLAQLVSFFPLQ*AIQSHYFLTTCHPSYSKILSKPCLILLGMWRIAL
QAPCTTSITLSVPSATAMAVALSKLTALFFSS*LQYCRRFHQYLRASHPLRPRSSTILVTIKSKSLKRW*LKRHRLRRSSQISAKDPFTNRMDLSLSLSP
PSSSIPSGLSGPLGVGRWV*NDNGEYPCLTLFTIESTAKTILKPTKPPTIIMNNFIYHSQFVMLNQFHKLAHLSNSNNSCSFFSCWFCDSSIKECIKLV*
LLISLSHSIQVV*FQICSRDLLLQLAFQGTAVVQMSFPEQPQIPRSC*SFRYLSTARILAWSCLMPQTVSCNRCCCASIALSKLFCCCTIPDNNCLACTV
SVIEAAPIVLPAAPKNPRNKAPIPTALFSLCTTLLFALVGATPNGSIFTTLYLYNSLLQLSLISPPPGLKISDSLLLLPPSLVNSSPVIFDEHLICPLMG
GAYIAFPTFCHMFIICFILHGRVIVSLPSVLFDPSVLQVLLNQVLLNSCVELQ*KNSPPQLKLCVTISGSPPEDCLKIIVLLFPNCSLNLLAICFKVLFH
FALLMLQCACLIFPIFPIVTNALPGPLWIRILFLVLLLGLVQLISTDVFSCTIMVLALSVKLTDLITTSSSARLPFNSS*VDTTGLIPCVHCTVLTFVHG
PVPLNVLLLHFRIAKPAGAQ*CMGIGSKDTFGQACVMTEVLQLVNL*LVVSLSIGIISSL*KNAYSFCTLPLMLVLILKEQFFISPFSIIILPLLLLVLV
SFFKSVHFKLTQSGVNFTHGFRL*SHKLIISSCICSTMSFFHMLKFSVTFTNTTSCGLGSVGTQACVAQTLCTSVSYALASDAQNRVVVASFHTGTP**T
VTHNFSVALQIINIPRSMVPHLHPHLHKC*YFSFTLIATVFCSFY*SIN*PVYFSLS*YFPIFYDYYGPHNYCYYYCYY*CYYCYYWYRLHYMYYLLL**
RSLMSLTVLMSSSSLSPLLPAIGDA*GFCYETNLAMKATLFTIAIGTSSFRLTSWMLPGL*SRIYWLHFLLSSVE*RLFCYVDTQF*KWINSSCCRILIM
ASTPAQVSP*VS*ICCPKPWSQILGKCLTASFLSSSKSSSVHSLCGSLCGPWSSGACSIYPLSVS*H*AKVALSFLVLLMLLVPSIVEILPCYVLLDIHT
*G*LCVLIRPFL*QSLKNSQSNTDELVGLLVQGLLVCYISFSSILWRLPDPNASLFLLYADPNMLLLPI*HPLVGCVLLNLFLDEGFHSDVYKTIP*LSL
KHTYGVLLNFSMF*SSSCLLATQSSPAICFP*SLMIFAFLLGTTFMSLLSCITTAPSPFQRSFAGPFQSGFLLSL**TRKF*IFVICFCNSLVCMSVAIM
STILSPALYPPIPPFLLKLWMNTAICTAVLRCSA*SLTCPIIFFNSLFIDSTTP*LWGL*GIPNSCLIPAHQQAALTVAPVKLLPLSVCIVFTGHLPANF
KRKYAVSCPVSAGITSASIYPLAT*TATRITFPSKCVQSSCHIPGLQSTCPCMASPFS*HLSQLATISFATTGGRLKSLAIALQLL*YFSCSSWALSIPS
KNSTFLIPALTNLSTCSFPPIPLCAGTHAR*TFSFFINCSII*LTNSDSL*SGCA*MIPNAYCESVTMFTSNPESCKAR*IACNSVF*FVVSVRVTTFCL
PLLVTYPAFPNLVSLLAAPST*KVSAPTMGSFSNWYHNFTKGGVLTNSHSGIQVACQYSVHHVSHVSFCMGSLNLGVFPHITMLSVVIFCTASVNCFTSL
VWAPLILAYFPVFRFLNGS**I*YVHWPCPCFCISAIKSFDGS*YTPCTGSFRISLFSASSSSASSVSGITSVSALVPLRSLHNCLTLIPG*I*LAQFNF
PTNFCMSLTVQLSFSGSTIGCTVHLSGWSS*PIQRNGGSF*CFLSGVVSPHLNRCCLSSSIFVLCCPISKSDPTYKSSMY**ITMSGFCFLKGSKIFVML
LWNIAGDPFHPCGSTLY*YLIPGVSLFILGMVNAVYFLKSSSKGTEKYASPTSSTVTDFFFFNPAGCGIPN*TSQKS*VLLLSSLKSTNFLHLVLSFFFM
ANTGVLYGFSGPIFEIFPSFSISVQISTNAFIFSSVNGHCLTFGPSIPGFNFTGTVSIGLMGKFKVQPI*VNRFLPIMLTGVGPTNTVPIALCPQISMSI
*SYCLTLIKPPIPPIIFGFHLPGKLISSNTVSSAPVSNRASFSCPPIFIVTRGRCQRVT*GKLKDTVPCLSAPASEGELLSLPQT*SSLLVGLLALVCSE
ENSLAFPCRKARSSLKN*PVSQYNLSFGVLPFHISNSPFS*GPCNFWLCALLCHN*NT*QSFFGS*NCLSASLW*LNLLLGSLLQPKLLPYGRVLLLPDM
LSSFLLV*PLVPMLLK*SYNLGSHFGPTRFLSSNFLPPVKLARLLEFYRTGLHSL*RVPLVLVLCPECW*GYTFLLFYLIPGLSIFYKFLLLG*VDYLSS
ILFVPEGY**FLLCHFPLVLSSGLVQ*ALHALDALYPILQLPH*WSLLTFAWLLDVPPLCLAWCLNLVGWLLLIMLKTWVSLLG*KPSLLLLLPMHLKF*
VIWPDVPFAPGCSAL*GNFG*PDCCVLCQLLLAVLFSYFCFALPLSCLKLPWCLLSLSFDAHNRGLLLYYIMI*VLLILSEGMVVAVPVFVYSLLMFLTG
QD*LRIVLAPCLPILYVLIYIFSFPLALTEFFPIDLILPRLILTLSHPSLSF*PPLVKIFGVLTSRRPSPLAVRASASRVLRRESSSGFPFAFRSLFGRH
C*RFSTLTKRV*GISSYQSHTTDGHTLLEALKASFIEA*AVGSLVSQRAPRLRSGLTRET

Esta secuencia contiene 105 subsecuencia(s):

<eliminado por no presentar relevancias>

El 6to marco de lectura genera la secuencia:

STQGKLY*GLSSGFPS*PESSQAQIWSNQRDPVQAKSSCLYAGSEGSPLPSPAQATPPWKVPSGKSLVASSMSAVLEVLRMQLSGHVMKC*AAVKPPL*H
FSLRVIHPMQAHRV*QAGVLSFIGLFYLIWLNWY*LVAPSKGQWISDPWPWCVVLLIREVALCVVDPQIKDILSSLGVN*PFQSPLFF*KVAKIYSCLVS
HWS*RYLRCDWKTHLLLLLCF*PGTSSIGSCCIATCDCSMFFQVSRCCSHPICCWLSSSHSFPYSRPSNHTTF*PLATHLIAKSFPSPVLFF*VCGE*LY
KLLVLLL*PYLSPQLLLWLWH*AS*QHYSLVPDSNTVGDSTNI*GLPTPCVPEVPQSSLQSRVSLSSGGS*RGTGSADRPR*VPRIRSLIEWICLCLSLH
LLLLFLRACRVPSGLGGGSETIMVNIPA*LYSL*KVQQKLFLNLPSLLLSL*IILYTTANLLC*TNSTNLPIYLIPIILVHSFLAGFAILQLRSVLSLCN
C*FLCPTPSRSCDSKSVPEIYYSN*HSKAQQWCK*VFQSNPKSPGAVDPLGIFPQPGFLPGAA*CPRL*VATDAVAPQ*PSANCSAAALYQTIIVWPVPS
ASLRLRP*CFLLLPRTQGTKLLFPLLFFLSAPLFSLPWWVLLLMVQFLLLYIYIIHFSNCPSYLLLQV*RSRTHCCYYHHLLLIAAL*YLMNI*FVH*WE
GHTLLFLLSATCL*FVLFCMGG*LCHFLQCYLTLQYSKYY*TKYY*TVVLNYSRKIPLHN*NCALQFLGPLLRIA*RLLFYYFQIVLLIC*LSVLKCYSI
LLY*CYNVLVSYFLFFLL*QMLSLVLSGYGFFFLYCCWVLYN*FLQMCSAVLLWF*HCP*N*QI*LLPLLLLDCHLTAVELILLA*FHVYIVLC*HLYMV
LFH*TSYYYILESQNQPGHNNVWELAQRIPLDRPV**LRCYNLSTYSW*YHYLLVLYQVYKKMHILSAPYLLCLC*Y*KSSFLSLLSPLSFSRYYYWY*Y
HSSNQCTLN*HRVGLILHMALGFDPIN*LYPHASVLPCHFSTC*NFLSHLPILLLVGWGLWVHRHVWPKHYVPLYHML*HLMHKIEWWLLPSTQVPHNRL
*PTIFL*HYRSSTSQGAWCPISTPISTSADISPSLSLPLSSALSISLSINLSIFLCLNIFLYSMITMDHTTIAIIIATTNATIATIGIGCITCTTYCFDR
EA**V*LF**ALRRCLRFFLP*EMPKAFVMKQTWQ*KQHFLQ*QLVQAVLG*LPGCFQGSSLGSTGSISCSPLSSNAYSAMSTPNSENG*TAVVAEFLLW
LPLLPKYPHKFHRYVALSHGAKS*ENV*QLHS*APLKALVSIHCVAPSVALGLLGLVPSILCQFRNTRQRWLYLFWCY*CC*CQVL*RSYLVMSCLIFTP
RANYVS**GLSYSRV*KTVKVIQMN*LVC*FRVYLCAISLFPPFYGDSLTQMPVSFSCMQTPICCYYQSSIP*WDVYF*TYSWMRAFIVMSIKPSPSFP*
NIHMVFY*TFPCSNPHPVYLPHNHHLPSVFHNP**SLLFFLALLLCHYYLVLLLPLHLSRGALLVLSKVDFCCPCNKPENFEFL*FVFVIL*FVCLLLLC
LLFFPLHCTPQSPLFF*NCG*ILPFVLLS*DVQPDLLPVL*FSLILYS*ILLLLDFGDCREFQIPA*FPPTNRRP*P*HR*NCCHCQYVLFLLAIFLLIL
KENMLFPALFLLE*LLLLYIHWLHELLPG*LFLLNVYNLVAIFLDYSLLVHAWLLLLADIYHSWLLFLLLLQVAG*NH*PLLSNYCDISHVHLGPYLFHL
KIVLS*FQH*LIYLLVHFLQFLCVLVPMPDRPFPFLLTALLFD*LTLIHFDLVVLE*FLMHIVSLLLCLLLIPNPAKLDKLLVTQSSDLLCQLG*QLFVF
LY**HILLFLI*SPC*LPHLHRRFLLLLWVLSLTGTIISLREGY*QTPTQESRWLANTLSTMFPMFPFVWAV*I*ESFPILLCFLWLFFALPLLIVLHH*
CGHPSFLHIFLFSDF*MALDKFDMSIGLAPASVFLLLSLLMGHNTLHVLVLLESLCFLPVLALLLLLVVLLLLVLWFL*GVYIIALL*SLGKSDLPNSIS
PLTSVCH*QSSCLFLAAL*AVLSIYQDGVHNPSKGMEVLSDVFCLVW*VPTSTDVVSAPLFLFYAALFLSQILHTNHPCIDR*LCLDFVF*KALRFLSCY
FGILLVILSIPVEAHCTDI*SLVSHCLY*VW*MQYTS*SLHLRELKNMHHPHPVLLLIFSFLTLRDVVFLIELPRSLEFSY*VL*NLLIFSI*YCLFSLW
QILEYCMDFQAQFLKFSLPFPSLYKFLLMLLFFLLSMAIV*LLGHPFLALILLVQSQ*G*WENLKCNQSESTDFFQLC*QV*VLLILYL*LYVHRFL*VS
DHTVLL**NLQFPLSFLVSIFLANSFLLILYHLLLYLIELPLVAPLSLL*RGVVAKE*PEGS*RIQFLVYRLLLLRGSCCLYPRPEALFWWGCWLWSALK
KIPWPSLVGRPDLP*KISLSLSTIFHLVSFLSTFPTALFPRGPAISGCVPFFATIETLNNLSLVPKIASLHHYGS*ICYLAHCFSQNSCLMAGSSYSLTC
CHHFF*CSRWSQCF*NSLTIWVRILDQQGFCHPIFYLL*SLLGS*SFIEPVYIVSKGFLWSLSYVQNAGRAIHSYYFI*SQDYPSFINFSYWDRWIICHP
SYLFLKGTSSSCYVTSPWFSHLAWCNRPCMHWMHSIPFCSFLIDGLF*HLHGCLMSPHCV*HGV*ILWGGSF**C*KHGYHFWAESLLFYYFYPCI*SSR
*YGLMYHLPLDVLHYRVILADLIAVSCVSCCLLCFFLTFVLLFLYLV*SFLGVFYLYPLMHTIEGCYCII**SKFF*SCLKGWL*LSQYLSTAF*CF*QA
RINCESF*LPACPYYMF*FIFFLSPWP*PNFFPSI*FSPA*Y*RSRTHLSPSSLR*SKFLAYSPVAAPRLLPCALQQAESCVERAPLVSLSLSGPCSGAT
ARDFPH*LKGSEGSLVTRVTQQTGTHYLKHSRQALLRLKQWVP*LARELPGSDLV*PER

Esta secuencia contiene 178 subsecuencia(s):

<eliminado por no presentar relevancias>

5.3.2 Simplificación de resultados de los análisis

En las siguientes tablas se muestran los resultados simplificados para los 10 análisis realizados. Las columnas muestran el gen localizado, su posición dentro de la secuencia de nucleótidos, el marco de lectura donde fue localizado y su producto final. Se indica dentro de paréntesis a la derecha del título de cada tabla el número del Código Genético utilizado para la traducción.

Tabla 6. Virus de inmunodeficiencia humana 1, genoma completo (1)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
gag-pol	336..1637	3°	Péptido de transición del marco Gag-Pol (1632..1637)
	1637..4642	2°	Péptido de transición del marco Gag-Pol (1637..1798)
			Pol (1655..4639)
			Proteasa (1799..2095)
			Transcriptasa inversa (2096..3775)
			Transcriptasa inversa, subunidad p51 (2096..3415)
			Integrasa (3776..4639)
gag	336..1838	3°	Matriz (336..731)
			Cápside (732..1424)
			p2 (1425..1466)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
			Nucleocápside (1467..1631)
			p1 (1632..1679)
			p6 (1680..1835)
vif	4587..5165	3°	Vif (4587..5165)
vpr	5105..5319	2°	Vpr (5105..5319, 5321..5396)
	5321..5396	3°	
tat	5377..5591	1°	Tat (5377..5591, 7925..7970)
	7925..7970	3°	
rev	5516..5591	2°	Rev (5516..5591, 7925..8199)
	7925..8199	1°	
vpu	5608..5856	1°	Vpu (5608..5856)
env	5771..8341	2°	Glicoproteína de envoltura superficial gp120 (5855..7303)
			Glicoproteína de envoltura transmembranal gp41 (7304..8338)
nef	8343..8963	3°	Nef (8343..8963) Excepción de traducción (pos:8712..8714, aa:Trp)
Duración del análisis: 00:00:55			

Tabla 7. DNA mitocondrial de *Chelonia mydas*, secuencia completa (2)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
ND1	2800..3770	1°	NADH deshidrogenasa, subunidad 1 (2800..3770) Excepción de traducción (pos:3769..3770, aa:TERM)
ND2	3978..5016	3°	NADH deshidrogenasa, subunidad 2 (3978..5016) Excepción de traducción (pos:5016, aa:TERM)
COX1	5401..6948	1°	Citocromo c oxidasa, subunidad I (5401..6948)
COX2	7083..7769	3°	Citocromo c oxidasa, subunidad II (7083..7769)
ATP8	7844..8029	2°	ATP sintetasa F0, subunidad 8 (7844..8029)
ATP6	7999..8681	1°	ATP sintetasa F0, subunidad 6 (7999..8681) Excepción de traducción (pos:8680..8681, aa:TERM)
COX3	8682..9465	3°	Citocromo c oxidasa, subunidad III (8682..9465) Excepción de traducción (pos:9465, aa:TERM)
ND3	9534..9707	3°	NADH deshidrogenasa, subunidad 3 (9534..9707, 9709..9883) Excepción de traducción (pos:9883, aa:TERM)
	9709..9883	1°	
ND4L	9954..10250	3°	NADH deshidrogenasa, subunidad 4L (9954..10250)
ND4	10244..11624	2°	NADH deshidrogenasa, subunidad 4 (10244..11624) Excepción de traducción (pos:11624, aa:TERM)
ND5	11832..13637	3°	NADH deshidrogenasa, subunidad 5 (11832..13637)
ND6	13633..14157	4°	NADH deshidrogenasa, subunidad 6 (13633..14157)
CYTB	14229..15372	3°	Citocromo b (14229..15372) Excepción de traducción (pos:15372, aa:TERM)
Duración del análisis: 00:02:45			

Tabla 8. Mitocondria de *Saccharomyces cerevisiae*, genoma completo (3)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
locus Q0010	3952..4338	1°	ORF hipotético (3952..4338)
locus Q0017	4254..4415	3°	ORF hipotético (4254..4415)
locus Q0032	11667..11957	3°	ORF hipotético (11667..11957)
COX1	13818..13986	3°	Citocromo c oxidasa, subunidad I (13818..13986, 16435..16470, 18954..18991, 20508..20984, 21995..22246, 23612..23746 ,25318..25342, 26229..26701)
	16435..16470		
	18954..18991	2°	
	20508..20984	3°	
	21995..22246	2°	
	23612..23746		
	25318..25342	1°	
	26229..26701	2°	
AI5_ALPHA	13818..13986	3°	Intrón de COX1 mitocondrial, al5-alpha (13818..13986, 16435..16470,

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
	16435..16470		18954..18991, 20508..20984, 21995..23167)
	18954..18991	2°	
	20508..20984	3°	
	21995..23167	2°	
AI4	13818..13986	3°	Intrón de COX1 mitocondrial (13818..13986, 18954..18991, 20508..21935)
	18954..18991	2°	
	20508..21935	3°	
AI3	13818..13986	3°	Intrón de COX1 mitocondrial, grupo móvil I (13818..13986, 18954..19996)
	18954..19996	2°	
AI2	13818..13986 16435..18830	3°	Intrón de COX1 mitocondrial, grupo móvil II (13818..13986, 16435..18830)
AI1	13818..16322	3°	Intrón de COX1 mitocondrial, grupo móvil II (13818..16322)
AI5_BETA	24156..24870	3°	Intrón de COX1 mitocondrial, aI5-beta (24156..24870, 24906..25255)
	24906..25255	2°	
AAP1	27666..27812	3°	ATP sintetasa mitocondrial, subunidad 8 (27666..27812)
ATP6	28487..29266	2°	ATP sintetasa mitocondrial, subunidad 6 (28487..29266)
locus Q0092	30874..31014	1°	ORF hipotético (30874..31014)
COB	36540..36954	3°	Citocromo b (36540..36954, 37723..37736, 39141..39217, 40841..41090, 42508..42558, 43297..43647)
	37723..37736		
	39141..39217		
	40841..41090		
	42508..42558	1°	
	43297..43647		
BI4	36540..36954	3°	mRNA maturasa mitocondrial bI4 (36540..36954, 37723..37736, 39141..39217, 40841..42251)
	37723..37736		
	39141..39217		
	40841..42251		
BI3	36540..36954	3°	mRNA maturasa mitocondrial bI3 (36540..36954, 37723..37736, 39141..40265)
	37723..37736		
	39141..40265		
BI2	36540..36954	3°	mRNA maturasa mitocondrial bI2 (36540..36954, 37723..38579)
	37723..38579		
OLI1	46723..46953	1°	ATP sintetasa mitocondrial, subunidad 9 (46723..46953)
VAR1	48901..50097	1°	Proteína ribosomal mitocondrial (48901..50097)
locus Q0142	51052..51228	1°	ORF hipotético (51052..51228)
locus Q0143	51277..51429	1°	ORF hipotético (51277..51429)
locus Q0144	54109..54438	1°	ORF hipotético (54109..54438)
SCEI	61022..61729	2°	DNA endonucleasa I-SceI (61022..61729)
locus Q0182	65770..66174	1°	ORF hipotético (65770..66174)
COX2	73758..74513	3°	Citocromo c oxidasa, subunidad II (73758..74513)
locus Q0255	74495..75622	2°	ORF hipotético (74495..75622, 75663..75872, 75904..75984)
	75663..75872	3°	
	75904..75984	1°	
COX3	79213..80022	1°	Citocromo c oxidasa, subunidad III (79213..80022)
Q0297	85554..85709	3°	ORF hipotético (85554..85709)
Duración del análisis: 00:15:04			

Tabla 9. Mitocondria de *Acanthamoeba castellanii*, genoma completo (4)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
rnl	2012..2440	2°	ORF142 (2012..2440)
	2849..3355		ORF168 (2849..3355)
	4207..4701	1°	ORF164 (4207..4701)
cox1/2	7353..9974	3°	Citocromo oxidasa, subunidad 1 y subunidad 2 (7353..9974)
rps4	10223..11347	2°	Proteína ribosomal S4 (10223..11347)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
nad6	11548..12330	1°	NADH deshidrogenasa, subunidad 6 (11548..12330)
nad5	12334..14361	1°	NADH deshidrogenasa, subunidad 5 (12334..14361)
nad11	14361..16388	3°	NADH deshidrogenasa, subunidad 11 (14361..16388)
cox3	16538..17527	2°	Citocromo oxidasa, subunidad 3 (16538..17527)
nad4	17537..19030	2°	NADH deshidrogenasa, subunidad 4 (17537..19030)
nad2	19035..20618	3°	NADH deshidrogenasa, subunidad 2 (19035..20618)
rps2	20635..21573	1°	Proteína ribosomal S2 (20635..21573)
atp9	22372..22611	1°	ATPasa transportadora de H(+), subunidad 9 (22372..22611)
cob	22743..23900	3°	Citocromo b (22743..23900)
nad4L	23900..24211	2°	NADH deshidrogenasa, subunidad 4L (23900..24211)
atp1	25416..26984	3°	ATPasa transportadora de H(+), subunidad 1 (25416..26984)
nad1	27010..28119	1°	NADH deshidrogenasa, subunidad 1 (27010..28119)
rpl11	28249..29268	1°	Proteína ribosomal L11 (28249..29268)
rps12	29234..29617	2°	Proteína ribosomal S12 (29234..29617)
rps7	29589..30602	3°	Proteína ribosomal S7 (29589..30602)
rpl2	30604..31365	1°	Proteína ribosomal L2 (30604..31365)
rps19	31373..31609	2°	Proteína ribosomal S19 (31373..31609)
rps3	31609..32505	1°	Proteína ribosomal S3 (31609..32505)
rpl16	32517..32942	3°	Proteína ribosomal L16 (32517..32942)
rpl14	32905..33294	1°	Proteína ribosomal L14 (32905..33294)
rpl5	33294..33827	3°	Proteína ribosomal L5 (33294..33827)
rps14	33830..34129	2°	Proteína ribosomal S14 (33830..34129)
rps8	34136..34519	2°	Proteína ribosomal S8 (34136..34519)
rpl6	34524..35069	3°	Proteína ribosomal L6 (34524..35069)
rps13	35084..35443	2°	Proteína ribosomal S13 (35084..35443)
rps11	35445..35966	3°	Proteína ribosomal S11 (35445..35966)
nad3	37125..37481	3°	NADH deshidrogenasa, subunidad 3 (37125..37481)
nad9	37472..38059	2°	NADH deshidrogenasa, subunidad 9 (37472..38059)
nad7	38041..39246	1°	NADH deshidrogenasa, subunidad 7 (38041..39246)
atp6	39250..39993	1°	ATPasa transportadora de H(+), subunidad 6 (39250..39993)
Duración del análisis: 00:04:55			

Tabla 10. Mitocondria de *Drosophila melanogaster*, genoma completo (5)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
ND2	240..1265	3°	NADH deshidrogenasa, subunidad 2 (240..1265)
COX1	<1474..3009	1°	Citocromo c oxidasa, subunidad I (<1474..3009)
COX2	3083..3767	2°	Citocromo c oxidasa, subunidad II (3083..3767) Excepción de traducción (pos:3767, aa:TERM)
ATP8	3907..4068	1°	ATP sintetasa F0, subunidad 8 (3907..4068)
ATP6	4062..4736	3°	ATP sintetasa F0, subunidad 6 (4062..4736)
COX3	4736..5524	2°	Citocromo c oxidasa, subunidad III (4736..5524)
ND3	5608..5961	1°	NADH deshidrogenasa, subunidad 3 (5608..5961)
ND5	6402..8124	6°	NADH deshidrogenasa, subunidad 5 (6402..8124) Excepción de traducción (pos:6402, aa:TERM)
ND4	8206..9544	5°	NADH deshidrogenasa, subunidad 4 (8206..9544) Excepción de traducción (pos:8206, aa:TERM)
ND4L	9544..9834	6°	NADH deshidrogenasa, subunidad 4L (9544..9834)
ND6	9970..10494	1°	NADH deshidrogenasa, subunidad 6 (9970..10494)
CYTB	10498..11634	1°	Citocromo b (10498..11634)
ND1	11720..12658	5°	NADH deshidrogenasa, subunidad 1 (11720..12658)
Duración del análisis: 00:03:06			

Tabla 11. mRNA para hemoglobina de *Tetrahymena pyriformis* (6)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
hemoglobina	17..382	2°	Hemoglobina (17..382)
Duración del análisis: 00:00:02			

Tabla 12. mRNA macronuclear para protein-cinasa nuclear putativa de *Euplotes octocarinatus* (gen npk 1) (10)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
npk 1	1..1272	1°	Protein-cinasa nuclear putativa (1..1272)
Duración del análisis: 00:00:06			

Tabla 13. Gen potenciador de la infectividad a macrófagos (mip) de *Legionella lytica* cepa LLAP-9, cds parciales (11)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
mip	<1..>598	2°	Potenciador de la infectividad a macrófagos (<1..>598)
Duración del análisis: 00:00:04			

Tabla 14. Mitocondria de *Scenedesmus obliquus*, genoma completo (22)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
orf76	770..1000	5°	ORF76 (770..1000)
orf130	3340..3732	5°	ORF130 (3340..3732)
cox2	4893..5342	3°	Citocromo oxidasa, subunidad 2 (4893..5342)
nad5	6092..6802 7414..8655	2°	NADH deshidrogenasa, subunidad 5 (6092..6802,7414..8655)
orf90	6853..7125	1°	ORF90 (6853..7125)
nad4L	9149..9451	3°	NADH deshidrogenasa, subunidad 4L (9149..9451)
nad3	10486..10839	2°	NADH deshidrogenasa, subunidad 3 (10486..10839)
cox3	11572..12507	2°	Citocromo oxidasa, subunidad 3 (11572..12507)
orf390	12718..13890	2°	ORF390 (12718..13890)
orf148	14345..14791	3°	ORF148 (14345..14791)
nad1	16227..17102	1°	NADH deshidrogenasa, subunidad 1 (16227..17102)
atp6	17783..18544	6°	ATPasa, subunidad 6 (17783..18544)
nad2	18971..20641	3°	NADH deshidrogenasa, subunidad 2 (18971..20641)
cox1	20938..22545	2°	Citocromo oxidasa, subunidad 1 (20938..22545)
nad4	24237..25802	1°	NADH deshidrogenasa, subunidad 4 (24237..25802)
orf151	29986..30441	2°	ORF151 (29986..30441)
nad6	33406..34065	2°	NADH deshidrogenasa, subunidad 6 (33406..34065)
cob	34106..35305	3°	Citocromo b (34106..35305)
atp9	35825..36046	3°	ATPasa, subunidad 9 (35825..36046)
orf367	36725..37828	3°	ORF367 (36725..37828)
Duración del análisis: 00:03:40			

Tabla 15. DNA mitocondrial de *Thraustochytrium aureum*, genoma parcial (23)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
nad7	1973..3166	2°	NADH deshidrogenasa, subunidad 7 (1973..3166)
rps8	7448..7825	2°	Proteína ribosomal S8 (7448..7825)
rpl6	7825..8343	1°	Proteína ribosomal L6 (7825..8343)
rps2	8353..8733	1°	Proteína ribosomal S2 (8353..8733)
rps4	8733..9203	3°	Proteína ribosomal S4 (8733..9203)
nad2	9247..10749	1°	NADH deshidrogenasa, subunidad 2 (9247..10749)
cox3	11168..11956	2°	Citocromo c oxidasa, subunidad 3 (11168..11956)
atp6	11966..12721	2°	ATP sintetasa F0, subunidad 6 (11966..12721)
rps14	12907..13155	3°	Proteína ribosomal S14 (12907..13155)
rpl14	13252..13620	3°	Proteína ribosomal L14 (13252..13620)
rpl5	13623..14195	2°	Proteína ribosomal L5 (13623..14195)
rps13	14192..14554	1°	Proteína ribosomal S13 (14192..14554)
rps11	14517..14873	2°	Proteína ribosomal S11 (14517..14873)
nad5	14976..16946	2°	NADH deshidrogenasa, subunidad 5 (14976..16946)
atp8	17065..17448	3°	ATP sintetasa F0, subunidad 8 (17065..17448)
ymf16	17560..18306	3°	Proteína transportadora independiente de SecY (17560..18306)
rpl2	18296..19006	1°	Proteína ribosomal L2 (18296..19006)
rps19	19006..19197	3°	Proteína ribosomal S19 (19006..19197)
rps3	19172..19927	1°	Proteína ribosomal S3 (19172..19927)
rpl16	19902..20327	2°	Proteína ribosomal L16 (19902..20327)

Gen	Posición	Marco de lectura (ProSA)	Producto (Posición)
cox2	20429..21181	1°	Citocromo c oxidasa, subunidad 2 (20429..21181)
cox1	21393..22904	2°	Citocromo c oxidasa, subunidad 1 (21393..22904)
nad4	22922..24388	1°	NADH deshidrogenasa, subunidad 4 (22922..24388)
nad11	24394..25035	3°	NADH deshidrogenasa, subunidad 11 (24394..25035)
nad1	25016..25981	1°	NADH deshidrogenasa, subunidad 1 (25016..25981)
nad4L	25994..26305	1°	NADH deshidrogenasa, subunidad 4L (25994..26305)
nad9	26484..27068	2°	NADH deshidrogenasa, subunidad 9 (26484..27068)
cob	27084..28226	2°	Apocitocromo b (27084..28226)
nad6	28410..29024	2°	NADH deshidrogenasa, subunidad 6 (28410..29024)
rps7	29024..29410	1°	Proteína ribosomal S7 (29024..29410)
rps12	29407..29769	3°	Proteína ribosomal S12 (29407..29769)
nad3	29867..30229	1°	NADH deshidrogenasa, subunidad 3 (29867..30229)
atp9	30348..30575	2°	ATP sintetasa F0, subunidad 9 (30348..30575)
Duración del análisis: 00:05:51			

6. Discusión

El análisis de secuencias es una de las metodologías más utilizadas en bioinformática. El desarrollo de herramientas computacionales adecuadas y eficientes permite auxiliar a los biólogos en la búsqueda del significado y función de las secuencias contenidas en los genes. El poder realizar estos análisis a través de Internet permite el acercamiento de todo tipo de usuarios, los cuales no requieren de conocimientos en programación para utilizar las aplicaciones.

ProSA es una aplicación con un diseño adecuado, pues éste permite un fácil mantenimiento y administración de la aplicación, además de que su instalación y configuración no es complicada. Otra de las principales ventajas de su diseño es que se pueden reutilizar todos sus módulos y funciones. Perl es un lenguaje que fomenta estas prácticas a través de su esquema de programación modular. La implementación puede utilizarse como base para el desarrollo de futuras aplicaciones, o extenderse para realizar búsquedas utilizando otras bases de datos u otro tipo de secuencias.

El resultado de esta implementación está orientado a computadoras con características similares (Linux, *BSD, Mac OS X y demás tipos de Unix), mas no por eso se encuentra limitado para su utilización en otras plataformas (Mac OS < 9, Windows) después de una configuración apropiada.

La utilización de Perl como lenguaje para desarrollar la implementación fue eficiente, esto se debió a que Perl es un lenguaje de programación suficientemente maduro, que cuenta con un gran soporte para la búsqueda de patrones y el desarrollo de aplicaciones Web. La gran cantidad de módulos disponibles a través de CPAN permite la rápida implementación de casi cualquier diseño de software.

A pesar de la existencia del proyecto BioPerl, no se utilizó para esta implementación ninguno de los módulos ahí existentes para la manipulación de secuencias, esto se debió a que -desde el punto de vista del autor- el proyecto BioPerl no es aún lo suficientemente maduro como para elaborar aplicaciones que se utilizarán dentro de un ambiente de producción (como es denominado en informática).

BioPerl funciona bien para aplicaciones desarrolladas dentro de un laboratorio, donde los usuarios son los mismos investigadores que elaboran herramientas para simplificar tareas diarias, mas no para usuarios finales que probablemente no puedan corregir los errores en las dependencias que ocasiona el continuo desarrollo y evolución de los módulos del proyecto.

Tomando en cuenta que con la tecnología de secuenciación actual solo es posible obtener secuencias de hasta 500 nucleótidos por experimento (Gibas *et al*., 2001), y que los tiempos obtenidos para los análisis con ProSA fueron bastante cortos (2 seg. para 587 nucleótidos), podemos decir que la aplicación es bastante eficiente para analizar secuencias obtenidas mediante estos experimentos.

Esto no significa que ProSA esté limitado para analizar secuencias mayores, por el contrario, la evidencia de que tal análisis es posible se presenta en los resultados aquí descritos (15:04 para 85779 nucleótidos). El único inconveniente será que para obtener una secuencia de tal longitud se necesitarán decenas de experimentos de secuenciación, además de un largo proceso previo de ensamblaje de la secuencia.

En el caso de que se contara con dicha secuencia, el análisis sería demasiado lento para llevarse a cabo por Internet. En el presente trabajo se realizaron modificaciones en la configuración de Apache, ajustando la variable `Timeout` a 1200 segundos (20 minutos), con lo cual el análisis de las secuencias grandes pudo llevarse a cabo. Esto se logró a costa de que la aplicación consumió una gran cantidad de memoria, debido al inmenso volumen de datos que fue generado, además de que el procesador era ocupado en un alto porcentaje por el proceso correspondiente. Esta aproximación no es la más adecuada, ya que en el caso de una máquina con pocos recursos (poca memoria y/o procesador lento),

estos dos factores podrían comprometer su rendimiento, sobre todo si ésta funciona como servidor de Web.

Una de las soluciones más adecuadas sería modificar el diseño de la aplicación orientándolo hacia un modelo de cómputo distribuido, en el que una computadora central se encargue de recibir las secuencias y se repartan fragmentos a otras maquinas para que realicen partes del análisis y devuelvan los resultados para su ensamble y despliegue al usuario (Loewe, 2002; Krieger *et al*., 2002). Este diseño podría reducir considerablemente el tiempo para la realización del análisis de una secuencia grande, además de que no agotaría los recursos de la máquina que funciona como servidor puesto que el trabajo estaría repartido.

ProSA es una aplicación sencilla para los usuarios nuevos y al mismo tiempo poderosa para los usuarios avanzados. La interfaz de Web resulta bastante fácil de utilizar, ya que cuenta con descripciones sobre la aplicación y su uso, así como vínculos a sitios que proporcionan información relacionada con los datos que se utilizan para el análisis.

El despliegue de los resultados se encuentra en un formato atractivo y legible para su interpretación. Una posible mejora sobre su aspecto podría ser el resaltar directamente en las secuencias de proteína los patrones de PROSITE encontrados. Se podrían consensar códigos de color para patrones que presenten actividad biológica semejante, tal como hacen los programas de visualización molecular, por ejemplo: *RasMol*.

Los resultados proporcionados por la aplicación son plenamente confiables, puesto que todos los patrones de PROSITE encontrados en los análisis coincidieron con la actividad correspondiente a los genes publicados en GenBank.

Ocurrieron algunas discrepancias en cuanto a los marcos de lectura donde se encontraron muchos de los patrones de actividad biológica significativa, las cuales son justificables debido a que la mayoría de las secuencias corresponden a organismos eucariotes. Es importante recordar que la estructura de los genes eucariotes es más complicada que la de los genes procariotes. Al contrario de los genes procariotes, los genes eucariotes se encuentran a menudo fragmentados en piezas que son ensambladas antes de la traducción. En eucariotes, el mRNA es procesado antes de ser traducido. Existen dos tipos de procesamiento: el corte y la poli-adenilación. El corte une las secuencias codificantes y elimina los elementos intermedios. Las secuencias que terminan dentro del mRNA maduro son llamadas exones, y las intermedias son llamadas intrones. En la poli-adenilación se añaden 50 o más nucleótidos de adenina al final del mRNA (Bedell *et al*., 2003). La siguiente figura ejemplifica el procesamiento del mRNA eucariótico:

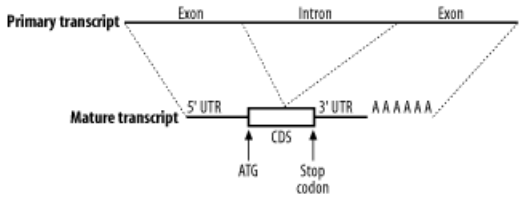


Figura 10. Procesamiento del mRNA eucariótico (Bedell *et al*., 2003)

ProSA es una aplicación que traduce las secuencias de nucleótidos tal y como son proporcionadas por el usuario. El único pre-procesamiento que lleva a cabo es el de eliminar los caracteres que no pertenezcan al tipo de secuencia seleccionado para el análisis, pero carece de métodos para la distinción de intrones y exones, por lo que la detección de secuencias no codificantes dependerá total y absolutamente de la habilidad y experiencia del usuario para la interpretación de los resultados.

Sin embargo, con la ayuda de ProSA será relativamente sencillo encontrar genes en genomas procariotes. Esto se debe a que los genes procariotes son un poco más simples. Estos contienen un promotor que determina cuando un gen debe ser transcrito y una región codificante que contiene la secuencia de DNA para una proteína. En estos genomas será muy raro encontrar marcos de lectura abiertos (ORF) largos, esto se debe a que es mas probable encontrar codones STOP cada 21 tripletes aproximadamente (ya que existen 3 codones STOP de un total de 64 combinaciones de tripletes). Por ejemplo, será realmente poco probable encontrar un ORF que tenga una longitud de 900 nucleótidos (en promedio, las proteínas poseen una longitud de 300 aminoácidos); aunque si sucediera, sería una clara señal de que el ORF codifica para una proteína real. Por supuesto, algunos genes codifican para proteínas pequeñas, y encontrar éstos será un poco más difícil (Bedell *et al*., 2003).

Por tales motivos, la interpretación de resultados de los análisis de secuencias pertenecientes a organismos eucariotes puede resultar un poco más compleja, sin embargo no es imposible, clara evidencia de esto es el presente trabajo.

La aplicación final se instaló como una herramienta en el Sitio Web de la Carrera de Biología (*http://biologia.iztacala.unam.mx/tools/prosa/*). Si se desea obtener una copia gratuita del software para instalación y uso local en otros servidores será necesario contactar al autor.

En conclusión, ProSA resulta una herramienta innovadora, puesto que no existía una aplicación que realizara búsquedas en la base de datos PROSITE a partir de secuencias de nucleótidos. Con el desarrollo de esta aplicación se ha podido resolver este problema, son los usuarios finales los que deberán juzgar los beneficios aquí planteados.

Apéndices

Apéndice A. Códigos Genéticos

1. Standard

```
TTT F Phe           TCT S Ser           TAT Y Tyr           TGT C Cys
TTC F Phe           TCC S Ser           TAC Y Tyr           TGC C Cys
TTA L Leu           TCA S Ser           TAA * Ter           TGA * Ter
TTG L Leu i         TCG S Ser           TAG * Ter           TGG W Trp
CTT L Leu           CCT P Pro           CAT H His           CGT R Arg
CTC L Leu           CCC P Pro           CAC H His           CGC R Arg
CTA L Leu           CCA P Pro           CAA Q Gln           CGA R Arg
CTG L Leu i         CCG P Pro           CAG Q Gln           CGG R Arg
ATT I Ile           ACT T Thr           AAT N Asn           AGT S Ser
ATC I Ile           ACC T Thr           AAC N Asn           AGC S Ser
ATA I Ile           ACA T Thr           AAA K Lys           AGA R Arg
ATG M Met i         ACG T Thr           AAG K Lys           AGG R Arg
GTT V Val           GCT A Ala           GAT D Asp           GGT G Gly
GTC V Val           GCC A Ala           GAC D Asp           GGC G Gly
GTA V Val           GCA A Ala           GAA E Glu           GGA G Gly
GTG V Val           GCG A Ala           GAG E Glu           GGG G Gly
```

2. Vertebrate Mitochondrial

```
TTT F Phe           TCT S Ser           TAT Y Tyr           TGT C Cys
TTC F Phe           TCC S Ser           TAC Y Tyr           TGC C Cys
TTA L Leu           TCA S Ser           TAA * Ter           TGA W Trp
TTG L Leu           TCG S Ser           TAG * Ter           TGG W Trp
CTT L Leu           CCT P Pro           CAT H His           CGT R Arg
CTC L Leu           CCC P Pro           CAC H His           CGC R Arg
CTA L Leu           CCA P Pro           CAA Q Gln           CGA R Arg
CTG L Leu           CCG P Pro           CAG Q Gln           CGG R Arg
ATT I Ile i         ACT T Thr           AAT N Asn           AGT S Ser
ATC I Ile i         ACC T Thr           AAC N Asn           AGC S Ser
ATA M Met i         ACA T Thr           AAA K Lys           AGA * Ter
ATG M Met i         ACG T Thr           AAG K Lys           AGG * Ter
GTT V Val           GCT A Ala           GAT D Asp           GGT G Gly
GTC V Val           GCC A Ala           GAC D Asp           GGC G Gly
GTA V Val           GCA A Ala           GAA E Glu           GGA G Gly
GTG V Val i         GCG A Ala           GAG E Glu           GGG G Gly
```

Tabla 16. Diferencias con respecto al Código Standard

Codón	Vertebrate Mitochondrial	Standard
AGA	* Ter	R Arg
AGG	* Ter	R Arg
AUA	M Met	I Ile
UGA	W Trp	* Ter

3. Yeast Mitochondrial

TTT F Phe	TCT S Ser	TAT Y Tyr	TGT C Cys
TTC F Phe	TCC S Ser	TAC Y Tyr	TGC C Cys
TTA L Leu	TCA S Ser	TAA * Ter	TGA W Trp
TTG L Leu	TCG S Ser	TAG * Ter	TGG W Trp
CTT T Thr	CCT P Pro	CAT H His	CGT R Arg
CTC T Thr	CCC P Pro	CAC H His	CGC R Arg
CTA T Thr	CCA P Pro	CAA Q Gln	CGA R Arg
CTG T Thr	CCG P Pro	CAG Q Gln	CGG R Arg
ATT I Ile	ACT T Thr	AAT N Asn	AGT S Ser
ATC I Ile	ACC T Thr	AAC N Asn	AGC S Ser
ATA M Met i	ACA T Thr	AAA K Lys	AGA R Arg
ATG M Met i	ACG T Thr	AAG K Lys	AGG R Arg
GTT V Val	GCT A Ala	GAT D Asp	GGT G Gly
GTC V Val	GCC A Ala	GAC D Asp	GGC G Gly
GTA V Val	GCA A Ala	GAA E Glu	GGA G Gly
GTG V Val	GCG A Ala	GAG E Glu	GGG G Gly

Nota: CGA y CGC están ausentes en este código genético.

Tabla 17. Diferencias con respecto al Código Standard

Codón	Yeast Mitochondrial	Standard
AUA	M Met	I Ile
CUU	T Thr	L Leu
CUC	T Thr	L Leu
CUA	T Thr	L Leu
CUG	T Thr	L Leu
UGA	W Trp	* Ter

4. Mold Mitochondrial, Protozoan Mitochondrial, Coelenterate Mitochondrial, Mycoplasma, Spiroplasma

TTT F Phe	TCT S Ser	TAT Y Tyr	TGT C Cys
TTC F Phe	TCC S Ser	TAC Y Tyr	TGC C Cys
TTA L Leu i	TCA S Ser	TAA * Ter	TGA W Trp
TTG L Leu i	TCG S Ser	TAG * Ter	TGG W Trp
CTT L Leu	CCT P Pro	CAT H His	CGT R Arg
CTC L Leu	CCC P Pro	CAC H His	CGC R Arg
CTA L Leu	CCA P Pro	CAA Q Gln	CGA R Arg
CTG L Leu i	CCG P Pro	CAG Q Gln	CGG R Arg
ATT I Ile i	ACT T Thr	AAT N Asn	AGT S Ser
ATC I Ile i	ACC T Thr	AAC N Asn	AGC S Ser
ATA I Ile i	ACA T Thr	AAA K Lys	AGA R Arg
ATG M Met i	ACG T Thr	AAG K Lys	AGG R Arg
GTT V Val	GCT A Ala	GAT D Asp	GGT G Gly
GTC V Val	GCC A Ala	GAC D Asp	GGC G Gly
GTA V Val	GCA A Ala	GAA E Glu	GGA G Gly
GTG V Val i	GCG A Ala	GAG E Glu	GGG G Gly

Tabla 18. Diferencias con respecto al Código Standard

Codón	Mold, Protozoan y Coelenterate Mitochondrial, Mycoplasma, Spiroplasma	Standard
UGA	W Trp	* Ter

5. Invertebrate Mitochondrial

```
TTT F Phe          TCT S Ser          TAT Y Tyr          TGT C Cys
TTC F Phe          TCC S Ser          TAC Y Tyr          TGC C Cys
TTA L Leu          TCA S Ser          TAA * Ter          TGA W Trp
TTG L Leu i        TCG S Ser          TAG * Ter          TGG W Trp
CTT L Leu          CCT P Pro          CAT H His          CGT R Arg
CTC L Leu          CCC P Pro          CAC H His          CGC R Arg
CTA L Leu          CCA P Pro          CAA Q Gln          CGA R Arg
CTG L Leu          CCG P Pro          CAG Q Gln          CGG R Arg
ATT I Ile i        ACT T Thr          AAT N Asn          AGT S Ser
ATC I Ile i        ACC T Thr          AAC N Asn          AGC S Ser
ATA M Met i        ACA T Thr          AAA K Lys          AGA S Ser
ATG M Met i        ACG T Thr          AAG K Lys          AGG S Ser
GTT V Val          GCT A Ala          GAT D Asp          GGT G Gly
GTC V Val          GCC A Ala          GAC D Asp          GGC G Gly
GTA V Val          GCA A Ala          GAA E Glu          GGA G Gly
GTG V Val i        GCG A Ala          GAG E Glu          GGG G Gly
```

Nota: El codón AGG esta ausente en *Drosophila*.

Tabla 19. Diferencias con respecto al Código Standard

Codón	Invertebrate Mitochondrial	Standard
AGA	S Ser	R Arg
AGG	S Ser	R Arg
AUA	M Met	I Ile
UGA	W Trp	* Ter

6. Ciliate Nuclear, Dasycladacean Nuclear, Hexamita Nuclear

```
TTT F Phe          TCT S Ser          TAT Y Tyr          TGT C Cys
TTC F Phe          TCC S Ser          TAC Y Tyr          TGC C Cys
TTA L Leu          TCA S Ser          TAA Q Gln          TGA * Ter
TTG L Leu          TCG S Ser          TAG Q Gln          TGG W Trp
CTT L Leu          CCT P Pro          CAT H His          CGT R Arg
CTC L Leu          CCC P Pro          CAC H His          CGC R Arg
CTA L Leu          CCA P Pro          CAA Q Gln          CGA R Arg
CTG L Leu          CCG P Pro          CAG Q Gln          CGG R Arg
ATT I Ile          ACT T Thr          AAT N Asn          AGT S Ser
ATC I Ile          ACC T Thr          AAC N Asn          AGC S Ser
ATA I Ile          ACA T Thr          AAA K Lys          AGA R Arg
ATG M Met i        ACG T Thr          AAG K Lys          AGG R Arg
GTT V Val          GCT A Ala          GAT D Asp          GGT G Gly
GTC V Val          GCC A Ala          GAC D Asp          GGC G Gly
GTA V Val          GCA A Ala          GAA E Glu          GGA G Gly
GTG V Val          GCG A Ala          GAG E Glu          GGG G Gly
```

Tabla 20. Diferencias con respecto al Código Standard

Codón	Ciliate Nuclear, Dasycladacean Nuclear, Hexamita Nuclear	Standard
UAA	Q Gln	* Ter
UAG	Q Gln	* Ter

9. Echinoderm Mitochondrial

```
TTT F Phe        TCT S Ser        TAT Y Tyr        TGT C Cys
TTC F Phe        TCC S Ser        TAC Y Tyr        TGC C Cys
TTA L Leu        TCA S Ser        TAA * Ter        TGA W Trp
TTG L Leu        TCG S Ser        TAG * Ter        TGG W Trp
CTT L Leu        CCT P Pro        CAT H His        CGT R Arg
CTC L Leu        CCC P Pro        CAC H His        CGC R Arg
CTA L Leu        CCA P Pro        CAA Q Gln        CGA R Arg
CTG L Leu        CCG P Pro        CAG Q Gln        CGG R Arg
ATT I Ile        ACT T Thr        AAT N Asn        AGT S Ser
ATC I Ile        ACC T Thr        AAC N Asn        AGC S Ser
ATA I Ile        ACA T Thr        AAA N Asn        AGA S Ser
ATG M Met i      ACG T Thr        AAG K Lys        AGG S Ser
GTT V Val        GCT A Ala        GAT D Asp        GGT G Gly
GTC V Val        GCC A Ala        GAC D Asp        GGC G Gly
GTA V Val        GCA A Ala        GAA E Glu        GGA G Gly
GTG V Val i      GCG A Ala        GAG E Glu        GGG G Gly
```

Tabla 21. Diferencias con respecto al Código Standard

Codón	Echinoderm Mitochondrial	Standard
AAA	N ASN	K Lys
AGA	S Ser	R Arg
AGG	S Ser	R Arg
UGA	W Trp	* Ter

10. Euplotid Nuclear

```
TTT F Phe        TCT S Ser        TAT Y Tyr        TGT C Cys
TTC F Phe        TCC S Ser        TAC Y Tyr        TGC C Cys
TTA L Leu        TCA S Ser        TAA * Ter        TGA C Cys
TTG L Leu        TCG S Ser        TAG * Ter        TGG W Trp
CTT L Leu        CCT P Pro        CAT H His        CGT R Arg
CTC L Leu        CCC P Pro        CAC H His        CGC R Arg
CTA L Leu        CCA P Pro        CAA Q Gln        CGA R Arg
CTG L Leu        CCG P Pro        CAG Q Gln        CGG R Arg
ATT I Ile        ACT T Thr        AAT N Asn        AGT S Ser
ATC I Ile        ACC T Thr        AAC N Asn        AGC S Ser
ATA I Ile        ACA T Thr        AAA K Lys        AGA R Arg
ATG M Met i      ACG T Thr        AAG K Lys        AGG R Arg
GTT V Val        GCT A Ala        GAT D Asp        GGT G Gly
GTC V Val        GCC A Ala        GAC D Asp        GGC G Gly
GTA V Val        GCA A Ala        GAA E Glu        GGA G Gly
GTG V Val        GCG A Ala        GAG E Glu        GGG G Gly
```

Tabla 22. Diferencias con respecto al Código Standard

Codón	Euplotid Nuclear	Standard
UGA	C Cys	* Ter

11. Bacterial y Plant Plastid

```
TTT F Phe           TCT S Ser           TAT Y Tyr           TGT C Cys
TTC F Phe           TCC S Ser           TAC Y Tyr           TGC C Cys
TTA L Leu           TCA S Ser           TAA * Ter           TGA * Ter
TTG L Leu i         TCG S Ser           TAG * Ter           TGG W Trp
CTT L Leu           CCT P Pro           CAT H His           CGT R Arg
CTC L Leu           CCC P Pro           CAC H His           CGC R Arg
CTA L Leu           CCA P Pro           CAA Q Gln           CGA R Arg
CTG L Leu i         CCG P Pro           CAG Q Gln           CGG R Arg
ATT I Ile i         ACT T Thr           AAT N Asn           AGT S Ser
ATC I Ile i         ACC T Thr           AAC N Asn           AGC S Ser
ATA I Ile i         ACA T Thr           AAA K Lys           AGA R Arg
ATG M Met i         ACG T Thr           AAG K Lys           AGG R Arg
GTT V Val           GCT A Ala           GAT D Asp           GGT G Gly
GTC V Val           GCC A Ala           GAC D Asp           GGC G Gly
GTA V Val           GCA A Ala           GAA E Glu           GGA G Gly
GTG V Val i         GCG A Ala           GAG E Glu           GGG G Gly
```

12. Alternative Yeast Nuclear

```
TTT F Phe           TCT S Ser           TAT Y Tyr           TGT C Cys
TTC F Phe           TCC S Ser           TAC Y Tyr           TGC C Cys
TTA L Leu           TCA S Ser           TAA * Ter           TGA * Ter
TTG L Leu           TCG S Ser           TAG * Ter           TGG W Trp
CTT L Leu           CCT P Pro           CAT H His           CGT R Arg
CTC L Leu           CCC P Pro           CAC H His           CGC R Arg
CTA L Leu           CCA P Pro           CAA Q Gln           CGA R Arg
CTG S Ser i         CCG P Pro           CAG Q Gln           CGG R Arg
ATT I Ile           ACT T Thr           AAT N Asn           AGT S Ser
ATC I Ile           ACC T Thr           AAC N Asn           AGC S Ser
ATA I Ile           ACA T Thr           AAA K Lys           AGA R Arg
ATG M Met i         ACG T Thr           AAG K Lys           AGG R Arg
GTT V Val           GCT A Ala           GAT D Asp           GGT G Gly
GTC V Val           GCC A Ala           GAC D Asp           GGC G Gly
GTA V Val           GCA A Ala           GAA E Glu           GGA G Gly
GTG V Val           GCG A Ala           GAG E Glu           GGG G Gly
```

Tabla 23. Diferencias con respecto al Código Standard

Codón	Alternative Yeast Nuclear	Standard
CUG	S Ser	L Leu

13. Ascidian Mitochondrial

```
TTT F Phe       TCT S Ser       TAT Y Tyr       TGT C Cys
TTC F Phe       TCC S Ser       TAC Y Tyr       TGC C Cys
TTA L Leu       TCA S Ser       TAA * Ter       TGA W Trp
TTG L Leu       TCG S Ser       TAG * Ter       TGG W Trp
CTT L Leu       CCT P Pro       CAT H His       CGT R Arg
CTC L Leu       CCC P Pro       CAC H His       CGC R Arg
CTA L Leu       CCA P Pro       CAA Q Gln       CGA R Arg
CTG L Leu       CCG P Pro       CAG Q Gln       CGG R Arg
ATT I Ile       ACT T Thr       AAT N Asn       AGT S Ser
ATC I Ile       ACC T Thr       AAC N Asn       AGC S Ser
ATA M Met       ACA T Thr       AAA K Lys       AGA G Gly
ATG M Met i     ACG T Thr       AAG K Lys       AGG G Gly
GTT V Val       GCT A Ala       GAT D Asp       GGT G Gly
GTC V Val       GCC A Ala       GAC D Asp       GGC G Gly
GTA V Val       GCA A Ala       GAA E Glu       GGA G Gly
GTG V Val       GCG A Ala       GAG E Glu       GGG G Gly
```

Tabla 24. Diferencias con respecto al Código Standard

Codón	Ascidian Mitochondrial	Standard
AGA	G Gly	R Arg
AGG	G Gly	R Arg
AUA	M Met	I Ile
UGA	W Trp	* Ter

14. Flatworm Mitochondrial

```
TTT F Phe       TCT S Ser       TAT Y Tyr       TGT C Cys
TTC F Phe       TCC S Ser       TAC Y Tyr       TGC C Cys
TTA L Leu       TCA S Ser       TAA Y Tyr       TGA W Trp
TTG L Leu       TCG S Ser       TAG * Ter       TGG W Trp
CTT L Leu       CCT P Pro       CAT H His       CGT R Arg
CTC L Leu       CCC P Pro       CAC H His       CGC R Arg
CTA L Leu       CCA P Pro       CAA Q Gln       CGA R Arg
CTG L Leu       CCG P Pro       CAG Q Gln       CGG R Arg
ATT I Ile       ACT T Thr       AAT N Asn       AGT S Ser
ATC I Ile       ACC T Thr       AAC N Asn       AGC S Ser
ATA I Ile       ACA T Thr       AAA N Asn       AGA S Ser
ATG M Met i     ACG T Thr       AAG K Lys       AGG S Ser
GTT V Val       GCT A Ala       GAT D Asp       GGT G Gly
GTC V Val       GCC A Ala       GAC D Asp       GGC G Gly
GTA V Val       GCA A Ala       GAA E Glu       GGA G Gly
GTG V Val       GCG A Ala       GAG E Glu       GGG G Gly
```

Tabla 25. Diferencias con respecto al Código Standard

Codón	Flatworm Mitochondrial	Standard
AAA	N Asn	K Lys
AGA	S Ser	R Arg
AGG	S Ser	R Arg
UAA	Y Tyr	* Ter
UGA	W Trp	* Ter

15. *Blepharisma* Macronuclear

```
TTT F Phe       TCT S Ser       TAT Y Tyr       TGT C Cys
TTC F Phe       TCC S Ser       TAC Y Tyr       TGC C Cys
TTA L Leu       TCA S Ser       TAA * Ter       TGA * Ter
TTG L Leu       TCG S Ser       TAG Q Gln       TGG W Trp
CTT L Leu       CCT P Pro       CAT H His       CGT R Arg
CTC L Leu       CCC P Pro       CAC H His       CGC R Arg
CTA L Leu       CCA P Pro       CAA Q Gln       CGA R Arg
CTG L Leu       CCG P Pro       CAG Q Gln       CGG R Arg
ATT I Ile       ACT T Thr       AAT N Asn       AGT S Ser
ATC I Ile       ACC T Thr       AAC N Asn       AGC S Ser
ATA I Ile       ACA T Thr       AAA K Lys       AGA R Arg
ATG M Met i     ACG T Thr       AAG K Lys       AGG R Arg
GTT V Val       GCT A Ala       GAT D Asp       GGT G Gly
GTC V Val       GCC A Ala       GAC D Asp       GGC G Gly
GTA V Val       GCA A Ala       GAA E Glu       GGA G Gly
GTG V Val       GCG A Ala       GAG E Glu       GGG G Gly
```

Tabla 26. Diferencias con respecto al Código Standard

Codón	*Blepharisma* Nuclear	Standard
UAG	Q Gln	* Ter

16. Chlorophycean Mitochondrial

```
TTT F Phe       TCT S Ser       TAT Y Tyr       TGT C Cys
TTC F Phe       TCC S Ser       TAC Y Tyr       TGC C Cys
TTA L Leu       TCA S Ser       TAA * Ter       TGA * Ter
TTG L Leu       TCG S Ser       TAG L Leu       TGG W Trp
CTT L Leu       CCT P Pro       CAT H His       CGT R Arg
CTC L Leu       CCC P Pro       CAC H His       CGC R Arg
CTA L Leu       CCA P Pro       CAA Q Gln       CGA R Arg
CTG L Leu       CCG P Pro       CAG Q Gln       CGG R Arg
ATT I Ile       ACT T Thr       AAT N Asn       AGT S Ser
ATC I Ile       ACC T Thr       AAC N Asn       AGC S Ser
ATA I Ile       ACA T Thr       AAA K Lys       AGA R Arg
ATG M Met i     ACG T Thr       AAG K Lys       AGG R Arg
GTT V Val       GCT A Ala       GAT D Asp       GGT G Gly
GTC V Val       GCC A Ala       GAC D Asp       GGC G Gly
GTA V Val       GCA A Ala       GAA E Glu       GGA G Gly
GTG V Val       GCG A Ala       GAG E Glu       GGG G Gly
```

Tabla 27. Diferencias con respecto al Código Standard

Codón	Chlorophycean Mitochondrial	Standard
UAG	L Leu	* Ter

21. Trematode Mitochondrial

```
TTT F Phe           TCT S Ser           TAT Y Tyr           TGT C Cys
TTC F Phe           TCC S Ser           TAC Y Tyr           TGC C Cys
TTA L Leu           TCA S Ser           TAA * Ter           TGA W Trp
TTG L Leu           TCG S Ser           TAG * Ter           TGG W Trp
CTT L Leu           CCT P Pro           CAT H His           CGT R Arg
CTC L Leu           CCC P Pro           CAC H His           CGC R Arg
CTA L Leu           CCA P Pro           CAA Q Gln           CGA R Arg
CTG L Leu           CCG P Pro           CAG Q Gln           CGG R Arg
ATT I Ile           ACT T Thr           AAT N Asn           AGT S Ser
ATC I Ile           ACC T Thr           AAC N Asn           AGC S Ser
ATA M Met           ACA T Thr           AAA N Asn           AGA S Ser
ATG M Met i         ACG T Thr           AAG K Lys           AGG S Ser
GTT V Val           GCT A Ala           GAT D Asp           GGT G Gly
GTC V Val           GCC A Ala           GAC D Asp           GGC G Gly
GTA V Val           GCA A Ala           GAA E Glu           GGA G Gly
GTG V Val i         GCG A Ala           GAG E Glu           GGG G Gly
```

Tabla 28. Diferencias con respecto al Código Standard

Codón	Trematode Mitochondrial	Standard
AAA	N Asn	K Lys
AGA	S Ser	R Arg
AGG	S Ser	R Arg
AUA	M Met	I Ile
UGA	W Trp	* Ter

22. *Scenedesmus obliquus* Mitochondrial

```
TTT F Phe           TCT S Ser           TAT Y Tyr           TGT C Cys
TTC F Phe           TCC S Ser           TAC Y Tyr           TGC C Cys
TTA L Leu           TCA * Ter           TAA * Ter           TGA * Ter
TTG L Leu           TCG S Ser           TAG L Leu           TGG W Trp
CTT L Leu           CCT P Pro           CAT H His           CGT R Arg
CTC L Leu           CCC P Pro           CAC H His           CGC R Arg
CTA L Leu           CCA P Pro           CAA Q Gln           CGA R Arg
CTG L Leu           CCG P Pro           CAG Q Gln           CGG R Arg
ATT I Ile           ACT T Thr           AAT N Asn           AGT S Ser
ATC I Ile           ACC T Thr           AAC N Asn           AGC S Ser
ATA I Ile           ACA T Thr           AAA K Lys           AGA R Arg
ATG M Met i         ACG T Thr           AAG K Lys           AGG R Arg
GTT V Val           GCT A Ala           GAT D Asp           GGT G Gly
GTC V Val           GCC A Ala           GAC D Asp           GGC G Gly
GTA V Val           GCA A Ala           GAA E Glu           GGA G Gly
GTG V Val           GCG A Ala           GAG E Glu           GGG G Gly
```

Tabla 29. Diferencias con respecto al Código Standard

Codón	*Scenedesmus obliquus* Mitochondrial	Standard
UAG	L Leu	* Ter
UCA	* Ter	S Ser

23. *Thraustochytrium* Mitochondrial

```
TTT F Phe          TCT S Ser          TAT Y Tyr          TGT C Cys
TTC F Phe          TCC S Ser          TAC Y Tyr          TGC C Cys
TTA * Ter          TCA S Ser          TAA * Ter          TGA * Ter
TTG L Leu          TCG S Ser          TAG * Ter          TGG W Trp
CTT L Leu          CCT P Pro          CAT H His          CGT R Arg
CTC L Leu          CCC P Pro          CAC H His          CGC R Arg
CTA L Leu          CCA P Pro          CAA Q Gln          CGA R Arg
CTG L Leu          CCG P Pro          CAG Q Gln          CGG R Arg
ATT I Ile i        ACT T Thr          AAT N Asn          AGT S Ser
ATC I Ile          ACC T Thr          AAC N Asn          AGC S Ser
ATA I Ile          ACA T Thr          AAA K Lys          AGA R Arg
ATG M Met i        ACG T Thr          AAG K Lys          AGG R Arg
GTT V Val          GCT A Ala          GAT D Asp          GGT G Gly
GTC V Val          GCC A Ala          GAC D Asp          GGC G Gly
GTA V Val          GCA A Ala          GAA E Glu          GGA G Gly
GTG V Val i        GCG A Ala          GAG E Glu          GGG G Gly
```

Tabla 30. Diferencias con respecto al Código Standard

Codón	*Thraustochytrium* Mitochondrial	Standard
UUA	* Ter	L Leu

Apéndice B. ProSA::Protein Sequence Analyzer

/home/prosa/bin/sync_PROSITE.pl

```
1   #!/usr/bin/perl -w
2   # sync_PROSITE.pl v1.0  22/10/2003
3   # Copyright © Mauricio Herrera Cuadra 2003
4   # web-biol@campus.iztacala.unam.mx
5   #
6   # Script para mantener sincronizada la base de datos PROSITE con la última versión disponible en:
7   # ftp://ftp.expasy.org/databases/prosite/release_with_updates/prosite.dat
8   #
9   # Colocar las siguientes 2 líneas en el crontab sustituyendo el correo por el del administrador del
10  # programa:
11  #
12  # MAILTO=USUARIO@SERVIDOR.DOMINIO
13  # 0 0 * * 1     /home/prosa/bin/sync_PROSITE.pl
14  #
15  # Modulos, pragmas y variables por utilizar
16  use strict;
17  use LWP::Simple;
18  use lib "/home/prosa/lib";
19  use LogUtils qw(write_log write_stdout);
20  use PROSITE qw(make_PROSITE_4_ProSA parse_PROSITE);
21
22  # Variables locales
23  my $app_dir = "/home/prosa";
24  my $log_file = "$app_dir/log/sync_PROSITE.log";
25  my $log_filehandle;
26  my $local_prosite = "$app_dir/var/prosite/prosite.dat";
27  my $nl = "\n";
28  my $parsed_prosite = "$app_dir/var/prosite/prosite_4_prosa.dat";
29  my $remote_prosite = "ftp://ftp.expasy.org/databases/prosite/release_with_updates/prosite.dat";
30  my $s = "||";
31
32  # Abre el archivo de log
33  open($log_filehandle, ">>", $log_file) or die("Imposible abrir el archivo de log: $log_file ($!)");
34
35  # Escribe en el log y en la salida estándar el inicio del proceso
36  write_log($log_filehandle, "notice", "La sincronización inició.");
37  write_stdout("notice", "La sincronización inició.");
38
39  # Sincroniza la base de datos PROSITE
40  my $rc = mirror($remote_prosite, $local_prosite);
41
42  # Revisa el status de la sincronización
43  if ($rc == RC_OK) {
44      write_log($log_filehandle, "notice", "Se descargó la base de datos PROSITE.");
45      write_stdout("notice", "Se descargó la base de datos PROSITE.");
46
47      # Obtiene los registros de PROSITE que se utilizarán para prosa.cgi y los almacena en un arreglo
48      my @PROSITE_patterns = parse_PROSITE($local_prosite, $s, $nl);
49      if (!@PROSITE_patterns) {
50          write_log($log_filehandle, "error", "Imposible accesar a la base de datos PROSITE:
    $local_prosite.");
51          write_stdout("error", "Imposible accesar a la base de datos PROSITE: $local_prosite.");
52          write_log($log_filehandle, "notice", "La sincronización finalizó inesperadamente.");
53          write_stdout("notice", "La sincronización finalizó inesperadamente.");
54          exit();
55      }
56
57      # Crea el archivo de destino prosite_4_prosa.dat
58      if (make_PROSITE_4_ProSA($parsed_prosite, @PROSITE_patterns)) {
59          write_log($log_filehandle, "notice", "Se creó el archivo destino: $parsed_prosite.");
60          write_stdout("notice", "Se creó el archivo destino: $parsed_prosite.");
```

```
61          } else {
62              write_log($log_filehandle, "error", "Imposible crear el archivo destino: $parsed_prosite.");
63              write_stdout("error", "Imposible crear el archivo destino: $parsed_prosite.");
64              write_log($log_filehandle, "notice", "La sincronización finalizó inesperadamente.");
65              write_stdout("notice", "La sincronización finalizó inesperadamente.");
66              exit();
67          }
68      } elsif ($rc == RC_NOT_MODIFIED) {
69          write_log($log_filehandle, "notice", "La base de datos PROSITE es actual.");
70          write_stdout("notice", "La base de datos PROSITE es actual.");
71      } else {
72          my $status = status_message($rc);
73          write_log($log_filehandle, "error", "Ocurrió el siguiente evento al sincronizar: $status($rc).");
74          write_stdout("error", "Ocurrió el siguiente evento al sincronizar: $status($rc).");
75          write_log($log_filehandle, "notice", "La sincronización finalizó inesperadamente.");
76          write_stdout("notice", "La sincronización finalizó inesperadamente.");
77          exit();
78      }
79
80      # Escribe en el log y en la salida estándar el final del proceso
81      write_log($log_filehandle, "notice", "La sincronización finalizó con éxito.");
82      write_stdout("notice", "La sincronización finalizó con éxito.");
83
84      # Cierra el archivo de log
85      close($log_filehandle) or die("Imposible cerrar el archivo de log: $log_file ($!)");
86
87      # Sale del programa
88      exit();
```

/home/prosa/cgi-bin/prosa.cgi

```
1     #!/usr/bin/perl -wT
2     # prosa.cgi v1.0    22/10/2003
3     # ProSA::Protein Sequence Analyzer
4     # Copyright © Mauricio Herrera Cuadra 2003
5     # web-biol@campus.iztacala.unam.mx
6     #
7     # Script CGI que analiza secuencias de nucleótidos o proteínas mediante una búsqueda de patrones
8     # conocidos en la base de datos PROSITE.
9     #
10    # Modulos, pragmas y variables por utilizar
11    use strict;
12    use CGI;
13    use Date::Calc qw(Delta_DHMS Today_and_Now);
14    use HTML::Template;
15    use lib "/home/prosa/lib";
16    use CodonTable qw(get_table_id get_trans_table);
17    use LogUtils qw(write_log);
18    use ProSA qw(analyze_protein);
19    use PROSITE qw(load_PROSITE);
20    use SeqStats qw(n_number n_percent seq_mw);
21    use SeqUtils qw(cut_seq DNA_2_mRNA make_orfs sanitize_seq translate);
22
23    # Variables locales
24    my $app_dir = "/home/prosa";
25    my $app_name = "ProSA";
26    my $app_version = "1.0";
27    my $client = $ENV{REMOTE_HOST} || $ENV{REMOTE_ADDR} || "UNKNOWN";
28    my $log_file = "$app_dir/log/prosa.log";
29    my $log_filehandle;
30    my $parsed_prosite = "$app_dir/var/prosite/prosite_4_prosa.dat";
31
32    # Abre el archivo de log
33    open($log_filehandle, ">>", $log_file) or die("Imposible abrir el archivo de log: $log_file ($!)");
34
35    # Escribe en el log el inicio del programa
36    write_log($log_filehandle, "notice", "[client $client] $app_name/$app_version inició.");
37
38    # Crea un nuevo objeto CGI
39    my $query = new CGI;
40
41    # Verifica la entrada de una secuencia y su tipo
42    my $sequence = $query->param('sequence') or printError("No se ha introducido una secuencia");
43    my $seq_type = $query->param('seq_type') or printError("No se ha seleccionado un tipo de secuencia");
44    ($seq_type eq "nucleotide" || $seq_type eq "aminoacid") or printError("El tipo de secuencia
      seleccionado no es válido");
45
46    # Realiza el análisis si la secuencia es de nucleótidos
47    if ($seq_type eq "nucleotide") {
48
49        # Escribe en el log la secuencia introducida
50        write_log($log_filehandle, "notice", "[client $client] La secuencia introducida fué:\n$sequence");
51
52        # Escribe en el log el tipo de secuencia
53        write_log($log_filehandle, "notice", "[client $client] El tipo de secuencia fué: $seq_type");
54
55        # Verifica la selección de una tabla de codones
56        my $table = $query->param('table') or printError("No se ha seleccionado un Código Genético para la
      traducción");
57
58        # Obtiene la tabla de codones que se utilizará para la traducción y su descripción
59        my $table_id = get_table_id($table) or printError("El Código Genético seleccionado para la
      traducción no existe");
60        my %trans_table = get_trans_table($table);
61
62        # Escribe en el log la tabla de codones seleccionada
```

```
63          write_log($log_filehandle, "notice", "[client $client] El Código Genético seleccionado fué:
    $table_id");
64
65          # Carga en memoria los datos del archivo de PROSITE parseado para prosa.cgi
66          my @PROSITE_patterns = load_PROSITE($parsed_prosite) or printError("Imposible accesar a la base de
    datos PROSITE");
67
68          # Escribe en el log el inicio del análisis
69          write_log($log_filehandle, "notice", "[client $client] El análisis comenzó.");
70
71          # Obtiene el tiempo al inicio del análisis
72          my @start = Today_and_Now();
73
74          # Valida la secuencia introducida y genera una nueva secuencia de mRNA
75          my $new_sequence = sanitize_seq($sequence, $seq_type) or printError("La secuencia de nucleótidos no
    es válida para el análisis");
76          my $mRNA_sequence = DNA_2_mRNA($new_sequence);
77
78          # Cuenta el número de nucleótidos y calcula el peso molecular de la secuencia
79          my $mRNA_seq_length = length($mRNA_sequence);
80          my $mRNA_seq_mw = seq_mw($mRNA_sequence, "nucleotide");
81
82          # Cuenta el número y calcula el porcentaje por cada nucleótido (A,U,G,C)
83          my %seq_stats = ();
84          for my $nucleotide qw(A U G C) {
85              $seq_stats{$nucleotide}{'number'} = n_number($mRNA_sequence, $nucleotide);
86              $seq_stats{$nucleotide}{'percent'} = n_percent($mRNA_sequence, $nucleotide);
87          }
88
89          # Calcula el porcentaje de Adenina-Uracilo
90          my $au_percent = $seq_stats{'A'}{'percent'} + $seq_stats{'U'}{'percent'};
91
92          # Calcula el porcentaje de Guanina-Citosina
93          my $gc_percent = $seq_stats{'G'}{'percent'} + $seq_stats{'C'}{'percent'};
94
95          # Crea 6 marcos de lectura para la secuencia
96          my @orf = make_orfs($mRNA_sequence);
97
98          # Realiza la traducción a proteína en cada marco de lectura
99          my @protein = ();
100         foreach my $orf(@orf) {
101             my $protein = translate($orf, %trans_table);
102             push(@protein, $protein);
103         }
104
105         # Analiza las proteinas obtenidas en cada marco de lectura
106         my @analyzed_proteins = ();
107         for my $i (0 .. 5) {
108             ($analyzed_proteins[$i]{'number'}, $analyzed_proteins[$i]{'sub_protein'}) =
    analyze_protein($protein[$i], \@PROSITE_patterns, 94);
109         }
110
111         # Obtiene el tiempo al final del análisis
112         my @end = Today_and_Now();
113
114         # Calcula la duración del análisis
115         my @lapse = Delta_DHMS(@start, @end);
116         my $elapsed_time = sprintf("%02d:%02d:%02d", $lapse[1], $lapse[2], $lapse[3]);
117
118         # Escribe en el log el final del análisis
119         write_log($log_filehandle, "notice", "[client $client] El análisis terminó.");
120
121         # Escribe en el log la duración del análisis
122         write_log($log_filehandle, "notice", "[client $client] La duración del análisis fué:
    $elapsed_time");
123
124         # Prepara las secuencias para su impresión en el navegador
125         for ($sequence, $mRNA_sequence, @protein) {
```

```
126             $_ = cut_seq($_, 100);
127         }
128 
129         # Crea un nuevo template y exporta las variables
130         my $template = HTML::Template->new(filename => "$app_dir/templates/nucleotide.tmpl");
131         $template->param(
132             sequence => $sequence,
133             elapsed_time => $elapsed_time,
134             mRNA_sequence => $mRNA_sequence,
135             mRNA_seq_length => $mRNA_seq_length,
136             mRNA_seq_mw => $mRNA_seq_mw,
137             a_number => $seq_stats{'A'}{'number'},
138             a_percent => sprintf("%.2f", $seq_stats{'A'}{'percent'}),
139             u_number => $seq_stats{'U'}{'number'},
140             u_percent => sprintf("%.2f", $seq_stats{'U'}{'percent'}),
141             g_number => $seq_stats{'G'}{'number'},
142             g_percent => sprintf("%.2f", $seq_stats{'G'}{'percent'}),
143             c_number => $seq_stats{'C'}{'number'},
144             c_percent => sprintf("%.2f", $seq_stats{'C'}{'percent'}),
145             au_percent => sprintf("%.2f", $au_percent),
146             gc_percent => sprintf("%.2f", $gc_percent),
147             table => $table,
148             table_id => $table_id,
149             protein_1 => $protein[0],
150             sp1_number => $analyzed_proteins[0]{'number'},
151             sub_protein_1 => $analyzed_proteins[0]{'sub_protein'},
152             protein_2 => $protein[1],
153             sp2_number => $analyzed_proteins[1]{'number'},
154             sub_protein_2 => $analyzed_proteins[1]{'sub_protein'},
155             protein_3 => $protein[2],
156             sp3_number => $analyzed_proteins[2]{'number'},
157             sub_protein_3 => $analyzed_proteins[2]{'sub_protein'},
158             protein_4 => $protein[3],
159             sp4_number => $analyzed_proteins[3]{'number'},
160             sub_protein_4 => $analyzed_proteins[3]{'sub_protein'},
161             protein_5 => $protein[4],
162             sp5_number => $analyzed_proteins[4]{'number'},
163             sub_protein_5 => $analyzed_proteins[4]{'sub_protein'},
164             protein_6 => $protein[5],
165             sp6_number => $analyzed_proteins[5]{'number'},
166             sub_protein_6 => $analyzed_proteins[5]{'sub_protein'},
167         );
168 
169         # Imprime el HTML de salida
170         print "Content-Type: text/html\n\n";
171         print $template->output;
172 
173         # Escribe en el log el final del programa
174         write_log($log_filehandle, "notice", "[client $client] $app_name/$app_version finalizó con
    éxito.");
175 
176         # Cierra el archivo de log
177         close($log_filehandle) or die("Imposible cerrar el archivo de log: $log_file ($!)");
178 
179         # Sale del programa
180         exit();
181 
182     # Realiza el análisis si la secuencia es de proteína
183     } elsif ($seq_type eq "aminoacid") {
184 
185         # Escribe en el log la secuencia introducida
186         write_log($log_filehandle, "notice", "[client $client] La secuencia introducida fué:\n$sequence");
187 
188         # Escribe en el log el tipo de secuencia
189         write_log($log_filehandle, "notice", "[client $client] El tipo de secuencia fué: $seq_type");
190 
191         # Carga en memoria los datos del archivo de PROSITE parseado para prosa.cgi
```

```
192     my @PROSITE_patterns = load_PROSITE($parsed_prosite) or printError("Imposible accesar a la base de
    datos PROSITE");
193 
194     # Escribe en el log el inicio del análisis
195     write_log($log_filehandle, "notice", "[client $client] El análisis comenzó.");
196 
197     # Obtiene el tiempo al inicio del análisis
198     my @start = Today_and_Now();
199 
200     # Valida la secuencia introducida y genera una nueva secuencia de proteína
201     my $protein = sanitize_seq($sequence, $seq_type) or printError("La secuencia de proteína no es
    válida para el análisis");
202 
203     # Analiza la proteína
204     my ($sp_number, $sub_protein_ref) = analyze_protein($protein, \@PROSITE_patterns, 94);
205 
206     # Obtiene el tiempo al final del análisis
207     my @end = Today_and_Now();
208 
209     # Calcula la duración del análisis
210     my @lapse = Delta_DHMS(@start, @end);
211     my $elapsed_time = sprintf("%02d:%02d:%02d", $lapse[1], $lapse[2], $lapse[3]);
212 
213     # Escribe en el log el final del análisis
214     write_log($log_filehandle, "notice", "[client $client] El análisis terminó.");
215 
216     # Escribe en el log la duración del análisis
217     write_log($log_filehandle, "notice", "[client $client] La duración del análisis fué:
    $elapsed_time");
218 
219     # Prepara las secuencias para su impresión en el navegador
220     for ($sequence, $protein) {
221         $_ = cut_seq($_, 100);
222     }
223 
224     # Crea un nuevo template y exporta las variables
225     my $template = HTML::Template->new(filename => "$app_dir/templates/aminoacid.tmpl");
226     $template->param(
227         sequence => $sequence,
228         elapsed_time => $elapsed_time,
229         protein => $protein,
230         sp_number => $sp_number,
231         sub_protein => $sub_protein_ref,
232     );
233 
234     # Imprime el HTML de salida
235     print "Content-Type: text/html\n\n";
236     print $template->output;
237 
238     # Escribe en el log el final del programa
239     write_log($log_filehandle, "notice", "[client $client] $app_name/$app_version finalizó con
    éxito.");
240 
241     # Cierra el archivo de log
242     close($log_filehandle) or die("Imposible cerrar el archivo de log: $log_file ($!)");
243 
244     # Sale del programa
245     exit();
246 }
247 
248 # Subrutina que imprime el error en caso de existir
249 sub printError {
250 
251     # Captura los parámetros enviados a la función
252     my ($error) = @_;
253 
254     # Crea un nuevo template y exporta las variables
255     my $template = HTML::Template->new(filename => "$app_dir/templates/error.tmpl");
```

```
256         $template->param(error => $error);
257 
258         # Imprime el HTML de salida
259         print "Content-Type: text/html\n\n";
260         print $template->output;
261 
262         # Escribe en el log el error y el final del programa
263         write_log($log_filehandle, "error", "[client $client] $error.");
264         write_log($log_filehandle, "notice", "[client $client] $app_name/$app_version finalizó
    inesperadamente.");
265 
266         # Cierra el archivo de log
267         close($log_filehandle) or die("Imposible cerrar el archivo de log: $log_file ($!)");
268 
269         # Sale del programa
270         exit();
271     }
```

/home/prosa/htdocs/prosa.html

```
1     <html>
2     <head>
3     <title>ProSA::Protein Sequence Analyzer</title>
4     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5     <meta http-equiv="Content-Language" content="es">
6     <meta http-equiv="Expires" content="0">
7     <meta http-equiv="Pragma" content="no-cache">
8     <meta http-equiv="Reply-to" content="web-biol@campus.iztacala.unam.mx">
9     <meta name="Author" content="Mauricio Herrera Cuadra">
10    <meta name="Description" content="Prosa::Protein Sequence Analyzer">
11    <meta name="Copyright" content="Copyright © Mauricio Herrera Cuadra 2003">
12    <meta name="Keywords" content="prosa, protein, sequence, analyzer, protein sequence analyzer">
13    </head>
14    <body>
15    <h1>ProSA::Protein Sequence Analyzer</h1>
16    <p><i>ProSA::Protein Sequence Analyzer</i> es una herramienta que busca dominios prote&iacute;nicos
      conocidos en secuencias de nucle&oacute;tidos o prote&iacute;na utilizando la <a
      href="http://www.expasy.org/prosite/" target="_blank">Base de Datos PROSITE</a>, con la finalidad de
      que el usuario pueda predecir a que posible familia de prote&iacute;nas pertenece una secuencia
      obtenida en el laboratorio.</p>
17    <hr width="100%">
18    <h2>Instrucciones de uso:</h2>
19    <ol>
20      <li>Teclea o pega (Ctrl+V) una secuencia de nucle&oacute;tidos o prote&iacute;na (los caracteres
      extra&ntilde;os ser&aacute;n eliminados).</li>
21      <li>Selecciona el tipo de secuencia.</li>
22      <li>Si la secuencia es de nucleotidos, selecciona el <a href="http://www.ncbi.nlm.nih.gov/htbin-
      post/Taxonomy/wprintgc?mode=c" target="_blank">C&oacute;digo Gen&eacute;tico</a> apropiado para su
      traducci&oacute;n.</li>
23      <li>Haz click en el bot&oacute;n &quot;Analizar&quot;.</li>
24    </ol>
25    <form action="/cgi-bin/prosa.cgi" method="post" name="ProSA">
26      <p>Teclea o pega tu secuencia aqu&iacute;:</p>
27      <textarea name="sequence" cols=80 rows=10 wrap="VIRTUAL"></textarea>
28      <p>Tu secuencia es: <input type=radio name=seq_type value="nucleotide"> DNA/RNA <input type=radio
      name=seq_type value="aminoacid"> Prote&iacute;na</p>
29      <p>Si tu secuencia es DNA/RNA, por favor selecciona el <a href="http://www.ncbi.nlm.nih.gov/htbin-
      post/Taxonomy/wprintgc?mode=c" target="_blank">C&oacute;digo Gen&eacute;tico</a> para la
      traducci&oacute;n:</p>
30      <select name="table">
31        <option value="1">1. Standard</option>
32        <option value="2">2. Vertebrate Mitochondrial</option>
33        <option value="3">3. Yeast Mitochondrial</option>
34        <option value="4">4. Mold Mitochondrial, Protozoan Mitochondrial, Coelenterate Mitochondrial,
      Mycoplasma, Spiroplasma</option>
35        <option value="5">5. Invertebrate Mitochondrial</option>
36        <option value="6">6. Ciliate Nuclear, Dasycladacean Nuclear, Hexamita Nuclear</option>
37        <option value="9">9. Echinoderm Mitochondrial</option>
38        <option value="10">10. Euplotid Nuclear</option>
39        <option value="11">11. Bacterial and Plant Plastid</option>
40        <option value="12">12. Alternative Yeast Nuclear</option>
41        <option value="13">13. Ascidian Mitochondrial</option>
42        <option value="14">14. Flatworm Mitochondrial</option>
43        <option value="15">15. Blepharisma Macronuclear</option>
44        <option value="16">16. Chlorophycean Mitochondrial</option>
45        <option value="21">21. Trematode Mitochondrial</option>
46        <option value="22">22. Scenedesmus obliquus mitochondrial</option>
47        <option value="23">23. Thraustochytrium mitochondrial code</option>
48      </select>
49      <br>
50      <br>
51      <input type="submit" value="Analizar">
52      <input type="reset" value="Limpiar">
53    </form>
```

```
54      <hr width="100%">
55      <p>Programa en <a href="http://www.perl.com/" target="_blank">Perl</a> por: <a href="mailto:web-
        biol@campus.iztacala.unam.mx">Mauricio Herrera Cuadra</a></p>
56      </body>
57      </html>
```

/home/prosa/lib/CodonTable.pm

```
1   # CodonTable.pm v1.0    22/10/2003
2   # Copyright © Mauricio Herrera Cuadra 2003
3   # web-biol@campus.iztacala.unam.mx
4   #
5   # Modulo basado en las tablas de uso de codones publicadas en la NCBI Genetic Codes home page:
6   # http://www.ncbi.nlm.nih.gov/htbin-post/Taxonomy/wprintgc?mode=c
7   #
8   # Declaración del nombre del modulo o paquete
9   package CodonTable;
10
11  # Modulos, pragmas y variables por utilizar
12  use strict;
13  use Exporter;
14  use vars qw(@ISA @EXPORT_OK);
15
16  # Arreglos de variables para la exportación de funciones y/o variables
17  @ISA = qw(Exporter);
18  @EXPORT_OK = qw(get_table_id get_trans_table);
19
20  # Subrutina     : get_table_id()
21  # Función       : Obtiene la descripción de la tabla de codones correspondiente al número
22  #                 enviado como parámetro o 'undef' en caso de que no exista la tabla
23  # Modo de uso   : $table_id = get_table_id("1")
24  # Parámetros    : $_[0] Un entero positivo ($table)
25  # Regresa       : $_[0] Una cadena con la descripción del número de tabla o 'undef' en caso de error
26  sub get_table_id {
27
28      # Captura los parámetros enviados a la función
29      my ($table) = @_;
30
31      # Hash con la descripción de cada tabla de codones
32      my %id = (
33          '1'  => 'Standard',
34          '2'  => 'Vertebrate Mitochondrial',
35          '3'  => 'Yeast Mitochondrial',
36          '4'  => 'Mold Mitochondrial, Protozoan Mitochondrial, Coelenterate Mitochondrial, Mycoplasma,
    Spiroplasma',
37          '5'  => 'Invertebrate Mitochondrial',
38          '6'  => 'Ciliate Nuclear, Dasycladacean Nuclear, Hexamita Nuclear',
39          '9'  => 'Echinoderm Mitochondrial',
40          '10' => 'Euplotid Nuclear',
41          '11' => 'Bacterial and Plant Plastid',
42          '12' => 'Alternative Yeast Nuclear',
43          '13' => 'Ascidian Mitochondrial',
44          '14' => 'Flatworm Mitochondrial',
45          '15' => 'Blepharisma Macronuclear',
46          '16' => 'Chlorophycean Mitochondrial',
47          '21' => 'Trematode Mitochondrial',
48          '22' => 'Scenedesmus obliquus mitochondrial',
49          '23' => 'Thraustochytrium mitochondrial code',
50      );
51
52      # Revisa que exista la descripción para la tabla seleccionada
53      if (exists $id{$table}) {
54
55          # Regresa los parámetros obtenidos por la función
56          return $id{$table};
57      } else {
58
59          # Regresa 'undef'
60          return;
61      }
62  }
63
64  # Subrutina     : get_trans_table()
```

```
65      # Función       : Obtiene los aminoácidos correspondientes a la tabla de codones cuyo número
66      #                 haya sido enviado como parámetro o 'undef' en caso de que no exista la tabla
67      # Modo de uso   : $trans_table = get_trans_table("1")
68      # Parámetros    : $_[0] Un entero positivo ($table)
69      # Regresa       : $_[0] Un hash con los aminoácidos de la tabla seleccionada o 'undef' en caso de
70      #                       error (%trans_table)
71      sub get_trans_table {
72
73          # Captura los parámetros enviados a la función
74          my ($table) = @_;
75
76          # Arreglo multidimensional con los aminoácidos correspondientes a cada tabla de codones
77          my @aa_array = (
78              [qw()],
79              [qw(F F L L S S S S Y Y * * C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
80              [qw(F F L L S S S S Y Y * * C C W W L L L L P P P P H H Q Q R R R R I I M M T T T T N N K K S S
        * * V V V V A A A A D D E E G G G G)],
81              [qw(F F L L S S S S Y Y * * C C W W T T T T P P P P H H Q Q R R R R I I M M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
82              [qw(F F L L S S S S Y Y * * C C W W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
83              [qw(F F L L S S S S Y Y * * C C W W L L L L P P P P H H Q Q R R R R I I M M T T T T N N K K S S
        S S V V V V A A A A D D E E G G G G)],
84              [qw(F F L L S S S S Y Y Q Q C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
85              [qw()],
86              [qw()],
87              [qw(F F L L S S S S Y Y * * C C W W L L L L P P P P H H Q Q R R R R I I I M T T T T N N N K S S
        S S V V V V A A A A D D E E G G G G)],
88              [qw(F F L L S S S S Y Y * * C C C W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
89              [qw(F F L L S S S S Y Y * * C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
90              [qw(F F L L S S S S Y Y * * C C * W L L L S P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
91              [qw(F F L L S S S S Y Y * * C C W W L L L L P P P P H H Q Q R R R R I I M M T T T T N N K K S S
        G G V V V V A A A A D D E E G G G G)],
92              [qw(F F L L S S S S Y Y Y * C C W W L L L L P P P P H H Q Q R R R R I I I M T T T T N N N K S S
        S S V V V V A A A A D D E E G G G G)],
93              [qw(F F L L S S S S Y Y * Q C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
94              [qw(F F L L S S S S Y Y * L C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
95              [qw()],
96              [qw()],
97              [qw()],
98              [qw()],
99              [qw(F F L L S S S S Y Y * * C C W W L L L L P P P P H H Q Q R R R R I I M M T T T T N N N K S S
        S S V V V V A A A A D D E E G G G G)],
100             [qw(F F L L S S * S Y Y * L C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
101             [qw(F F * L S S S S Y Y * * C C * W L L L L P P P P H H Q Q R R R R I I I M T T T T N N K K S S
        R R V V V V A A A A D D E E G G G G)],
102         );
103
104         # Revisa que exista el arreglo de aminoácidos para la tabla seleccionada
105         if (defined @{$aa_array[$table]}) {
106             my @aa = @{$aa_array[$table]};
107
108             # Hash con las 64 posibles combinaciones de codones para la traducción
109             my %trans_table = (
110                 'UUU' => $aa[0],  'UUC' => $aa[1],  'UUA' => $aa[2],  'UUG' => $aa[3],
111                 'UCU' => $aa[4],  'UCC' => $aa[5],  'UCA' => $aa[6],  'UCG' => $aa[7],
112                 'UAU' => $aa[8],  'UAC' => $aa[9],  'UAA' => $aa[10], 'UAG' => $aa[11],
113                 'UGU' => $aa[12], 'UGC' => $aa[13], 'UGA' => $aa[14], 'UGG' => $aa[15],
114                 'CUU' => $aa[16], 'CUC' => $aa[17], 'CUA' => $aa[18], 'CUG' => $aa[19],
115                 'CCU' => $aa[20], 'CCC' => $aa[21], 'CCA' => $aa[22], 'CCG' => $aa[23],
```

```
116             'CAU' => $aa[24], 'CAC' => $aa[25], 'CAA' => $aa[26], 'CAG' => $aa[27],
117             'CGU' => $aa[28], 'CGC' => $aa[29], 'CGA' => $aa[30], 'CGG' => $aa[31],
118             'AUU' => $aa[32], 'AUC' => $aa[33], 'AUA' => $aa[34], 'AUG' => $aa[35],
119             'ACU' => $aa[36], 'ACC' => $aa[37], 'ACA' => $aa[38], 'ACG' => $aa[39],
120             'AAU' => $aa[40], 'AAC' => $aa[41], 'AAA' => $aa[42], 'AAG' => $aa[43],
121             'AGU' => $aa[44], 'AGC' => $aa[45], 'AGA' => $aa[46], 'AGG' => $aa[47],
122             'GUU' => $aa[48], 'GUC' => $aa[49], 'GUA' => $aa[50], 'GUG' => $aa[51],
123             'GCU' => $aa[52], 'GCC' => $aa[53], 'GCA' => $aa[54], 'GCG' => $aa[55],
124             'GAU' => $aa[56], 'GAC' => $aa[57], 'GAA' => $aa[58], 'GAG' => $aa[59],
125             'GGU' => $aa[60], 'GGC' => $aa[61], 'GGA' => $aa[62], 'GGG' => $aa[63],
126         );
127
128         # Regresa los parámetros obtenidos por la función
129         return %trans_table;
130     } else {
131
132         # Regresa 'undef'
133         return;
134     }
135 }
136
137 1;
```

/home/prosa/lib/LogUtils.pm

```
1     # LogUtils.pm v1.0    22/10/2003
2     # Copyright © Mauricio Herrera Cuadra 2003
3     # web-biol@campus.iztacala.unam.mx
4     #
5     # Modulo que se encarga de manipular diferentes tipos de mensajes.
6     #
7     # Declaración del nombre del modulo o paquete
8     package LogUtils;
9
10    # Modulos, pragmas y variables por utilizar
11    use strict;
12    use Exporter;
13    use vars qw(@ISA @EXPORT_OK);
14
15    # Arreglos de variables para la exportación de funciones y/o variables
16    @ISA = qw(Exporter);
17    @EXPORT_OK = qw(write_log write_stdout);
18
19    # Subrutina     : write_log()
20    # Función       : Escribe un mensaje con la fecha y el tipo de mensaje en un archivo de log
21    # Modo de uso   : write_log($log_filehandle, $msg_type, $msg)
22    # Parámetros    : $_[0] Una cadena con el filehandle del archivo de log ($log_filehandle)
23    #                 $_[1] Una cadena con el tipo de mensaje que se escribirá ($msg_type)
24    #                 $_[2] Una cadena con el mensaje que se escribirá ($msg)
25    # Regresa       : $_[0] 'true' en caso de éxito o muere en caso de error
26    sub write_log {
27
28        # Captura los parámetros enviados a la función
29        my ($log_filehandle, $msg_type, $msg) = @_;
30
31        # Obtiene la fecha actual
32        my $date = localtime();
33
34        # Imprime en el archivo de log o muere en caso de error
35        print $log_filehandle "[$date] [$msg_type] $msg\n" or die("Imposible escribir en el archivo de
      log.");
36    }
37
38    # Subrutina     : write_stdout()
39    # Función       : Escribe un mensaje con la fecha y el tipo de mensaje en la salida estándar
40    # Modo de uso   : write_stdout($msg_type, $msg)
41    # Parámetros    : $_[0] Una cadena con el tipo de mensaje que se escribirá ($msg_type)
42    #                 $_[1] Una cadena con el mensaje que se escribirá ($msg)
43    # Regresa       : $_[0] 'true' en caso de éxito o muere en caso de error
44    sub write_stdout {
45
46        # Captura los parámetros enviados a la función
47        my ($msg_type, $msg) = @_;
48
49        # Obtiene la fecha actual
50        my $date = localtime();
51
52        # Imprime en la salida estándar o muere en caso de error
53        print "[$date] [$msg_type] $msg\n" or die("Imposible escribir en la salida estándar.");
54    }
55
56    1;
```

/home/prosa/lib/ProSA.pm

```
1     # ProSA.pm v1.0    22/10/2003
2     # Copyright © Mauricio Herrera Cuadra 2003
3     # web-biol@campus.iztacala.unam.mx
4     #
5     # Modulo que se encarga del análisis de proteínas dentro de "prosa.cgi".
6     #
7     # Declaración del nombre del modulo o paquete
8     package ProSA;
9
10    # Modulos, pragmas y variables por utilizar
11    use strict;
12    use Exporter;
13    use lib "/home/prosa/lib";
14    use PROSITE qw(search_PROSITE);
15    use SeqStats qw(seq_mw);
16    use SeqUtils qw(cut_seq split_by_stops);
17    use vars qw(@ISA @EXPORT_OK);
18
19    # Arreglos de variables para la exportación de funciones y/o variables
20    @ISA = qw(Exporter);
21    @EXPORT_OK = qw(analyze_protein);
22
23    # Subrutina     : analyze_protein()
24    # Función       : Analiza una secuencia de proteína por el siguiente método:
25    #                 - La separa en subsecuencias en caso de que contenga 'stops(*)' (@sub_proteins)
26    #                 - Cuenta el número de subsecuencias obtenidas ($sp_number)
27    #                 - Crea un arreglo (@sub_protein) en donde se almacenarán por cada subsecuencia los
28    #                   siguientes datos en forma de un hash (%row):
29    #                   * Secuencia de proteína ($sub_proteins)
30    #                   * Longitud de la secuencia ($aa_number)
31    #                   * Peso molecular ($prot_seq_mw)
32    #                   * El arreglo (@PROSITE_search) que es el resultado de la búsqueda en PROSITE
33    #                     realizada mediante la función search_PROSITE() definida en PROSITE.pm
34    # Modo de uso   : ($sp_number, $sub_prot_ref) = analyze_protein($protein, \@PROSITE_patterns[, $cut])
35    # Parámetros    : $_[0] Una cadena que deberá ser una secuencia de proteína previamente validada por
36    #                       la función sanitize_seq() definida en SeqUtils.pm ($protein)
37    #                 $_[1] Una referencia hacia el arreglo de patrones obtenidos por la función
38    #                       load_PROSITE() definida en PROSITE.pm (\@PROSITE_patterns)
39    #                 $_[2] (Opcional) Un entero con la longitud del párrafo deseada para la impresión
40    #                       de los resultados ($cut)
41    # Regresa       : $_[0] Un entero con el número de subsecuencias obtenidas ($sp_number)
42    #                 $_[1] Una referencia hacia los resultados de la búsqueda en PROSITE para todas las
43    #                       subsecuencias (@sub_protein)
44    sub analyze_protein {
45
46        # Captura los parámetros enviados a la función
47        my ($protein, $PROSITE_patterns_ref, $cut) = @_;
48        my @PROSITE_patterns = @$PROSITE_patterns_ref;
49
50        # Obtiene subsecuencias a partir de la proteína
51        my @sub_proteins = split_by_stops($protein);
52
53        # Cuenta el número de subsecuencias obtenidas
54        my $sp_number = scalar(@sub_proteins);
55
56        # Inicializa el arreglo que se devuelve como segundo parámetro
57        my @sub_protein = ();
58
59        # Analiza cada subsecuencia
60        foreach my $sub_proteins(@sub_proteins) {
61
62            # Mide la longitud de la subsecuencia en turno
63            my $aa_number = length($sub_proteins);
64
65            # Obtiene el peso molecular de la subsecuencia en turno
```

```
66              my $prot_seq_mw = seq_mw($sub_proteins, "aminoacid");
67
68              # Obtiene el resultado de la búsqueda en PROSITE de la subsecuencia en turno
69              my @PROSITE_search = search_PROSITE($sub_proteins, @PROSITE_patterns);
70
71              # Añade un salto de línea cada 'n' caracteres (si la subsecuencia en turno los excede)
72              # NOTA: Esta opción solo es necesaria si los resultados se van a imprimir en la pantalla de
73              #       un navegador Web. Lo que hace es evitar un scroll horizontal mayor que la pantalla
74              #       del navegador y mejorar el formato de salida.
75              $sub_proteins = cut_seq($sub_proteins, $cut) if defined $cut;
76
77              # Crea el hash con los resultados del análisis
78              my %row = (
79                  'sub_protein' => $sub_proteins,
80                  'aa_number' => $aa_number,
81                  'prot_seq_mw' => $prot_seq_mw,
82                  'PROSITE_search' => [@PROSITE_search],
83              );
84
85              # Empuja el hash como referencia dentro del arreglo que se va a devolver
86              push(@sub_protein, \%row);
87          }
88
89          # Regresa los parámetros obtenidos por la función
90          return $sp_number, \@sub_protein;
91      }
92
93      1;
```

/home/prosa/lib/PROSITE.pm

```
1     # PROSITE.pm v1.0    22/10/2003
2     # Copyright © Mauricio Herrera Cuadra 2003
3     # web-biol@campus.iztacala.unam.mx
4     #
5     # Modulo que se encarga de diferentes tareas respecto a la base de datos PROSITE.
6     #
7     # Declaración del nombre del modulo o paquete
8     package PROSITE;
9
10    # Modulos, pragmas y variables por utilizar
11    use strict;
12    use Exporter;
13    use vars qw(@ISA @EXPORT_OK);
14
15    # Arreglos de variables para la exportación de funciones y/o variables
16    @ISA = qw(Exporter);
17    @EXPORT_OK = qw(load_PROSITE make_PROSITE_4_ProSA parse_PROSITE search_PROSITE);
18
19    # Subrutina     : get_line_types()
20    # Función       : Traduce un registro de PROSITE en un hash con los "tipos de linea"
21    # Modo de uso   : $line_types_ref = get_line_types($record)
22    # Parámetros    : $_[0] Una cadena que deberá ser un registro de PROSITE completo ($record)
23    # Regresa       : $_[0] Una referencia hacia el hash con los "tipos de línea" para el registro
24    #                       introducido (%line_types)
25    sub get_line_types {
26
27        # Captura los parámetros enviados a la función
28        my ($record) = @_;
29
30        # Inicia el hash donde se almacenarán los "tipos de línea"
31        my %line_types = ();
32
33        # Separa en cada salto de línea el registro enviado como parámetro y lo almacena en un arreglo
34        my @records = split(/\n/, $record);
35
36        # Recorre cada elemento del arreglo
37        foreach my $records(@records) {
38
39            # Extrae las 2 primeras letras para definir el "tipo de línea"
40            my $type = substr($records, 0, 2);
41
42            # Verifica si ya existe el "tipo de línea" como llave en el hash %line_types
43            if (defined $line_types{$type}) {
44
45                # Si existe, le concatena el elemento en turno
46                $line_types{$type} .= $records;
47            } else {
48
49                # Si no existe, define el "tipo de línea" como nueva llave y le da el valor del elemento
50                # en turno
51                $line_types{$type} = $records;
52            }
53        }
54
55        # Regresa los parámetros obtenidos por la función
56        return \%line_types;
57    }
58
59    # Subrutina     : load_PROSITE()
60    # Función       : Elabora un arreglo a partir de los registros contenidos en el archivo de PROSITE
61    #                 parseado para "prosa.cgi". Cada elemento del arreglo corresponde a un registro de
62    #                 PROSITE, que a su vez esta separado dentro de un hash cuyas llaves corresponden al
63    #                 "tipo de línea"
64    # Modo de uso   : @PROSITE_patterns = load_PROSITE($parsed_prosite)
65    # Parámetros    : $_[0] Una cadena que contiene la ruta hacia el archivo de PROSITE parseado para
```

```
66      #                       "prosa.cgi" ($parsed_prosite)
67      # Regresa       : $_[0] Un arreglo con los registros de PROSITE que se utilizarán para la búsqueda
68      #                       por "prosa.cgi" o 'undef' en caso de error (@PROSITE_patterns)
69      sub load_PROSITE {
70
71          # Captura los parámetros enviados a la función
72          my ($parsed_prosite) = @_;
73
74          # Crea un filehandle para el archivo de PROSITE parseado para "prosa.cgi"
75          my $parse_filehandle;
76
77          # Abre el archivo enviado como parámetro o regresa 'undef' en caso de error
78          open($parse_filehandle, "<", $parsed_prosite) or return;
79
80          # Inicia el arreglo donde se almacenará el hash correspondiente a cada registro
81          my @PROSITE_patterns = ();
82
83          # Ejecuta un ciclo que extrae una línea por cada vuelta hasta encontrar el fin del archivo
84          while (my $record = <$parse_filehandle>) {
85
86              # Elimina los saltos de línea del registro en turno
87              chomp $record;
88
89              # Inicia el hash donde se almacenarán los "tipos de línea" del registro en turno
90              my %pattern = ();
91
92              # Separa en cada '||' el registro en turno en las llaves correspondientes al "tipo de línea"
93              ($pattern{'id'}, $pattern{'accession'}, $pattern{'description'}, $pattern{'pattern'},
    $pattern{'regexp'}) = split(/\|\|/, $record);
94
95              # Empuja el hash dentro del arreglo principal
96              push(@PROSITE_patterns, {%pattern});
97          }
98
99          # Cierra el archivo o regresa 'undef' en caso de error
100         close($parse_filehandle) or return;
101
102         # Regresa los parámetros obtenidos por la función
103         return @PROSITE_patterns;
104     }
105
106     # Subrutina     : make_PROSITE_4_ProSA()
107     # Función       : Crea el archivo de PROSITE parseado para "prosa.cgi" 'prosite_4_prosa.dat'
108     # Modo de uso   : make_PROSITE_4_ProSA($parsed_prosite, @PROSITE_patterns)
109     # Parámetros    : $_[0] Una cadena que contiene la ruta hacia el archivo de PROSITE parseado para
110     #                       "prosa.cgi" ($parsed_prosite)
111     #                 $_[1] El arreglo generado por la función parse_PROSITE() (@PROSITE_patterns)
112     # Regresa       : $_[0] 'true' en caso de éxito o 'undef' en caso de error
113     sub make_PROSITE_4_ProSA {
114
115         # Captura los parámetros enviados a la función
116         my ($parsed_prosite, @PROSITE_patterns) = @_;
117
118         # Crea un filehandle para el archivo de destino
119         my $parse_filehandle;
120
121         # Abre el archivo de destino 'prosite_4_prosa.dat' o regresa 'undef' en caso de error
122         open($parse_filehandle, ">", $parsed_prosite) or return;
123
124         # Imprime en el archivo los registros contenidos en el arreglo @PROSITE_patterns o regresa
125         # 'undef' en caso de error
126         print $parse_filehandle @PROSITE_patterns or return;
127
128         # Cierra el archivo o regresa 'undef' en caso de error
129         close($parse_filehandle) or return;
130     }
131
132     # Subrutina     : parse_PROSITE()
```

```perl
# Función       : Abre la base de datos PROSITE y extrae unicamente los registros que se van a
#                 utilizar para "prosa.cgi"
# Modo de uso   : @PROSITE_patterns = parse_PROSITE($prosite_file, $s, $nl)
# Parámetros    : $_[0] Una cadena que contiene la ruta hacia la base de datos PROSITE local
#                       ($prosite_file)
#                 $_[1] Un caracter o cadena que será el nuevo separador de los "tipos de línea" de
#                       un registro ($s)
#                 $_[2] Un caracter o cadena que será el nuevo separador de los registros ($nl)
# Regresa       : $_[0] Un arreglo con los registros que se utilizarán para "prosa.cgi" o 'undef' en
#                       caso de error (@PROSITE_patterns)
sub parse_PROSITE {

    # Captura los parámetros enviados a la función
    my ($prosite_file, $s, $nl) = @_;

    # Crea un filehandle para la base de datos PROSITE local
    my $prosite_filehandle;

    # Abre el archivo enviado como parámetro o regresa 'undef' en caso de error
    open($prosite_filehandle, "<", $prosite_file) or return;

    # Inicia el arreglo donde se almacenarán los registros que se van a utilizar
    my @PROSITE_patterns = ();

    # Establece "//\n" como el separador por defecto de la entrada de datos
    $/ = "//\n";

    # Ejecuta un ciclo que extrae un registro por cada vuelta hasta encontrar el fin del archivo.
    # Ejemplo de un registro de PROSITE (patrón):
    #   ID   ASN_GLYCOSYLATION; PATTERN.
    #   AC   PS00001;
    #   DT   APR-1990 (CREATED); APR-1990 (DATA UPDATE); APR-1990 (INFO UPDATE).
    #   DE   N-glycosylation site.
    #   PA   N-{P}-[ST]-{P}.
    #   CC   /TAXO-RANGE=??E?V;
    #   CC   /SITE=1,carbohydrate;
    #   CC   /SKIP-FLAG=TRUE;
    #   DO   PDOC00001;
    #   //
    while (my $record = <$prosite_filehandle>) {

        # Obtiene un hash con los "tipos de línea" para el registro en turno
        my $line_types_ref = get_line_types($record);
        my %line_types = %$line_types_ref;

        # Si existe el "tipo de línea" 'ID' revisa que contenga 'PATTERN' para continuar el ciclo,
        # de lo contrario salta al siguiente registro
        next unless defined $line_types{'ID'} and $line_types{'ID'} =~ /PATTERN/;

        # Extrae el valor del "tipo de línea" 'ID' y lo procesa
        my $id = $line_types{'ID'};
        $id =~ s/^ID   //;
        $id =~ s/; .*//;

        # Extrae el valor del "tipo de línea" 'AC' y lo procesa
        my $accession = $line_types{'AC'};
        $accession =~ s/^AC   //;
        $accession =~ s/;$//;

        # Extrae el valor del "tipo de línea" 'DE' y lo procesa
        my $description = $line_types{'DE'};
        $description =~ s/DE   //g;
        $description =~ s/\.$//;

        # Extrae el valor del "tipo de línea" 'PA' y lo procesa
        my $pattern = $line_types{'PA'};
        $pattern =~ s/PA   //g;
        $pattern =~ s/\.$//;
```

```
201
202             # Elabora una expresión regular de Perl a partir de $pattern
203             my $regexp = PROSITE_2_regexp($pattern);
204
205             # Elabora un nuevo registro para el arreglo principal
206             my $PROSITE_pattern = "$id$s$accession$s$description$s$pattern$s$regexp$nl";
207
208             # Empuja el registro dentro del arreglo principal
209             push(@PROSITE_patterns, $PROSITE_pattern);
210         }
211
212         # Cierra el archivo o regresa 'undef' en caso de error
213         close($prosite_filehandle) or return;
214
215         # Regresa los parámetros obtenidos por la función
216         return @PROSITE_patterns;
217     }
218
219     # Subrutina     : PROSITE_2_regexp()
220     # Función       : Calcula una expresión regular de Perl a partir de un patrón de PROSITE
221     # Modo de uso   : $regexp = PROSITE_2_regexp($pattern)
222     # Parámetros    : $_[0] Una cadena que contiene el patrón de PROSITE ($pattern)
223     # Regresa       : $_[0] Una cadena con una expresión regular equivalente al patrón de PROSITE
224     #                       ($regexp)
225     sub PROSITE_2_regexp {
226
227         # Captura los parámetros enviados a la función
228         my ($regexp) = @_;
229
230         # Convierte algunos caracteres especiales a su equivalente en expresión regular
231         $regexp =~ s/{/[^/g;
232         $regexp =~ tr/cx}()<>\-\./C.]{}^$/d;
233         $regexp =~ s/\[G\$\]/(G|\$)/;
234
235         # Regresa los parámetros obtenidos por la función
236         return $regexp;
237     }
238
239     # Subrutina     : search_PROSITE()
240     # Función       : Busca patrones de la base de datos PROSITE dentro de una secuencia de proteína
241     # Modo de uso   : @PROSITE_search = search_PROSITE($protein, @PROSITE_patterns)
242     # Parámetros    : $_[0] Una cadena que contiene una secuencia de proteína previamente validada por
243     #                       la función sanitize_seq() definida en SeqUtils.pm ($protein)
244     #                 $_[1] El arreglo generado por la función load_PROSITE() (@PROSITE_patterns)
245     # Regresa       : $_[0] Un arreglo con los resultados de la búsqueda para esa secuencia de proteína
246     #                       o Cero '0' si no se producen resultados (@PROSITE_search)
247     sub search_PROSITE {
248
249         # Captura los parámetros enviados a la función
250         my ($protein, @PROSITE_patterns) = @_;
251
252         # Inicia el arreglo donde se almacenarán los resultados de la búsqueda
253         my @PROSITE_search = ();
254
255         # Ejecuta un ciclo que recorre cada elemento del arreglo @PROSITE_patterns enviado como
256         # parámetro
257         for my $i (0 .. $#PROSITE_patterns) {
258
259             # Extrae del elemento en turno la expresión regular que se utilizará para la búsqueda
260             my $regexp = $PROSITE_patterns[$i]{'regexp'};
261
262             # Ejecuta un ciclo cuando la expresión regular sea encontrada dentro de la secuencia de
263             # proteína enviada como parámetro
264             while ($protein =~ /$regexp/g) {
265
266                 # Inicia el hash donde se almacenarán los "tipos de línea" correspondientes a la
267                 # expresión regular coincidente
268                 my %search = ();
```

```
269
270             # Extrae los valores de cada "tipo de línea" y los asigna dentro del hash
271             $search{'id'} = $PROSITE_patterns[$i]{'id'};
272             $search{'accession'} = $PROSITE_patterns[$i]{'accession'};
273             $search{'description'} = $PROSITE_patterns[$i]{'description'};
274             $search{'pattern'} = $PROSITE_patterns[$i]{'pattern'};
275             $search{'regexp'} = $regexp;
276
277             # Asigna dentro del hash la posición de la coincidencia
278             $search{'position'} = pos($protein) - length($&) + 1;
279
280             # Asigna dentro del hash el texto coincidente con la expresión regular
281             $search{'match'} = $&;
282
283             # Empuja el hash dentro del arreglo principal
284             push(@PROSITE_search, {%search});
285         }
286     }
287
288     # Regresa los parámetros obtenidos por la función.
289     return @PROSITE_search;
290 }
291
292 1;
```

/home/prosa/lib/SeqStats.pm

```
1     # SeqStats.pm v1.0    22/10/2003
2     # Copyright © Mauricio Herrera Cuadra 2003
3     # web-biol@campus.iztacala.unam.mx
4     #
5     # Modulo que se encarga de obtener información a partir de secuencias.
6     #
7     # Declaración del nombre del modulo o paquete
8     package SeqStats;
9
10    # Modulos, pragmas y variables por utilizar
11    use strict;
12    use Exporter;
13    use vars qw(@ISA @EXPORT_OK);
14
15    # Arreglos de variables para la exportación de funciones y/o variables
16    @ISA = qw(Exporter);
17    @EXPORT_OK = qw(n_number n_percent seq_mw);
18
19    # Hash que contiene los pesos moleculares de diferentes tipos de moleculas
20    my %mw = (
21        'nucleotide' => {
22            'A' => '329.24',    # Adenina
23            'U' => '306.19',    # Uracilo
24            'G' => '345.24',    # Guanina
25            'C' => '305.21',    # Citosina
26        },
27        'aminoacid' => {
28            'A' => '89.09',     # Alanina
29            'C' => '121.15',    # Cisteína
30            'D' => '133.1',     # Ácido Aspártico
31            'E' => '147.13',    # Ácido Glutámico
32            'F' => '165.19',    # Fenilalanina
33            'G' => '75.07',     # Glicina
34            'H' => '155.16',    # Histidina
35            'I' => '131.18',    # Isoleucina
36            'K' => '146.19',    # Lisina
37            'L' => '131.18',    # Leucina
38            'M' => '149.22',    # Metionina
39            'N' => '132.12',    # Asparagina
40            'P' => '115.13',    # Prolina
41            'Q' => '146.15',    # Glutamina
42            'R' => '174.21',    # Arginina
43            'S' => '105.09',    # Serina
44            'T' => '119.12',    # Treonina
45            'V' => '117.15',    # Valina
46            'W' => '204.22',    # Triptófano
47            'Y' => '181.19',    # Tirosina
48        },
49    );
50
51    # Subrutina     : n_number()
52    # Función       : Cuenta la cantidad de un nucleótido determinado dentro de una secuencia
53    # Modo de uso   : $n_number = n_number($seq, $nuc)
54    # Parámetros    : $_[0] Una cadena que contiene una secuencia de nucleótidos previamente validada
55    #                       por la función sanitize_seq() definida en SeqUtils.pm ($seq)
56    #                 $_[1] Una cadena con el nucleótido que se quiere contar (ej. "A") ($nuc)
57    # Regresa       : $_[0] Un entero con la cantidad del nucleótido enviado como parámetro ($n_number)
58    sub n_number {
59
60        # Captura los parámetros enviados a la función
61        my ($seq, $nuc) = @_;
62
63        # Elimina todos los caracteres que no sean el nucleótido enviado como parámetro
64        $seq =~ s/[^$nuc]//g;
65
```

```
66          # Regresa la longitud de la cadena resultante
67          return length($seq);
68      }
69
70      # Subrutina     : n_percent()
71      # Función       : Calcula el porcentaje de un nucleótido determinado dentro de una secuencia
72      # Modo de uso   : $n_percent = n_percent($n_number, $seq_length)
73      # Parámetros    : $_[0] Una cadena que contiene una secuencia de nucleótidos previamente validada
74      #                       por la función sanitize_seq() definida en SeqUtils.pm ($seq)
75      #                 $_[1] Una cadena con el nucleótido del que se quiere obtener porcentaje (ej. "A")
76      #                       ($nuc)
77      # Regresa       : $_[0] Una cadena con el porcentaje del nucleótido enviado como parámetro
78      #                       ($n_percent)
79      sub n_percent {
80
81          # Captura los parámetros enviados a la función
82          my ($seq, $nuc) = @_;
83
84          # Elimina todos los caracteres que no sean el nucleótido enviado como parámetro
85          my $n = $seq;
86          $n =~ s/[^$nuc]//g;
87
88          # Regresa el porcentaje del nucleótido dentro de la secuencia
89          return (length($n) * 100) / length($seq);
90      }
91
92      # Subrutina     : seq_mw()
93      # Función       : Calcula el peso molecular de una secuencia
94      # Modo de uso   : $seq_mw = seq_mw($seq)
95      # Parámetros    : $_[0] Una cadena que contiene una secuencia previamente validada por la función
96      #                       sanitize_seq() definida en SeqUtils.pm ($seq)
97      #                 $_[1] Una cadena que contiene el tipo de secuencia (ej. "nucleotide") ($type)
98      # Regresa       : $_[0] Una cadena con el peso molecular de la secuencia o muere en caso de error
99      #                       ($seq_mw)
100     sub seq_mw {
101
102         # Captura los parámetros enviados a la función
103         my ($seq, $type) = @_;
104
105         # Revisa que el tipo de secuencia enviado como parámetro sea válido o muere en caso de error
106         ($type eq "nucleotide" || $type eq "aminoacid") or die("El tipo de secuencia enviado no es
    válido");
107
108         # Inicia el entero donde se sumarán los pesos moleculares
109         my $seq_mw = 0;
110
111         # Ejecuta un ciclo por cada letra de la secuencia
112         for my $i (0 .. length($seq)) {
113
114             # Extrae de la secuencia la letra en turno
115             my $elem = substr($seq, $i, 1);
116
117             # Si la letra existe en el hash de pesos moleculares correspondiente a su tipo, suma el peso
118             # molecular de la letra al entero que se va a regresar.
119             $seq_mw += $mw{$type}{$elem} if defined $mw{$type}{$elem};
120         }
121
122         # Regresa los parámetros obtenidos por la función
123         return $seq_mw;
124     }
125
126     1;
```

/home/prosa/lib/SeqUtils.pm

```
1   # SeqUtils.pm v1.0    22/10/2003
2   # Copyright © Mauricio Herrera Cuadra 2003
3   # web-biol@campus.iztacala.unam.mx
4   #
5   # Modulo que se encarga de diferentes tareas con secuencias.
6   #
7   # Declaración del nombre del modulo o paquete
8   package SeqUtils;
9
10  # Modulos, pragmas y variables por utilizar
11  use strict;
12  use Exporter;
13  use vars qw(@ISA @EXPORT_OK);
14
15  # Arreglos de variables para la exportación de funciones y/o variables
16  @ISA = qw(Exporter);
17  @EXPORT_OK = qw(cut_seq DNA_2_mRNA make_orfs sanitize_seq split_by_stops translate);
18
19  # Subrutina     : cut_seq()
20  # Función       : Añade saltos de línea a una secuencia si esta excede del número de caracteres
21  #                 indicado. Esta función solo es necesaria si los resultados se van a imprimir en la
22  #                 pantalla de un navegador Web. Lo que hace es evitar un scroll horizontal mayor que
23  #                 la pantalla del navegador y mejorar el formato.
24  # Modo de uso   : $seq = cut_seq($seq, $num_char)
25  # Parámetros    : $_[0] Una cadena que contiene una secuencia de nucleótidos o proteína previamente
26  #                       validada por la función sanitize_seq() ($seq)
27  #                 $_[1] Un entero que será el límite para añadir los saltos de línea ($num_char)
28  # Regresa       : $_[0] La cadena enviada como parámetro con los saltos de línea insertados en la
29  #                       posición solicitada ($seq)
30  sub cut_seq {
31
32      # Captura los parámetros enviados a la función
33      my ($seq, $num_char) = @_;
34
35      # Si la secuencia excede del número de caracteres enviado como parámetro añade un salto de línea
36      # al final de cada subcadena
37      $seq =~ s/(.{$num_char})/$1\n/g;
38
39      # Regresa los parámetros obtenidos por la función
40      return $seq;
41  }
42
43  # Subrutina     : DNA_2_mRNA()
44  # Función       : Convierte una secuencia de DNA en mRNA
45  # Modo de uso   : $mRNA = DNA_2_mRNA($seq)
46  # Parámetros    : $_[0] Una cadena que contiene una secuencia de nucleótidos previamente validada
47  #                       por la función sanitize_seq() ($seq)
48  # Regresa       : $_[0] Una cadena de mRNA ($mRNA)
49  sub DNA_2_mRNA {
50
51      # Captura los parámetros enviados a la función
52      my ($mRNA) = @_;
53
54      # Sustituye todas las Timinas(T) por Uracilos(U)
55      $mRNA =~ tr/T/U/;
56
57      # Regresa los parámetros obtenidos por la función
58      return $mRNA;
59  }
60
61  # Subrutina     : make_orfs()
62  # Función       : Crea 6 marcos de lectura para la traducción a partir de una secuencia de mRNA
63  # Modo de uso   : @orf = make_orfs($mRNA)
64  # Parámetros    : $_[0] Una cadena que contiene una secuencia de mRNA generada por la función
65  #                       DNA_2_mRNA() ($mRNA)
```

```
66      # Regresa       : $_[0] Un arreglo de 6 elementos donde cada uno es un marco de lectura listo para
67      #                       la traducción a proteína (@orf)
68      sub make_orfs {
69
70          # Captura los parámetros enviados a la función
71          my ($seq) = @_;
72
73          # Copia 6 veces dentro del arreglo @orf la secuencia enviada como parámetro
74          my @orf = ($seq, $seq, $seq, $seq, $seq, $seq);
75
76          # Elimina el primer caracter del segundo elemento del arreglo
77          $orf[1] =~ s/^.//;
78
79          # Elimina los dos primeros caracteres del tercer elemento del arreglo
80          $orf[2] =~ s/^.{2}//;
81
82          # Invierte la cadena que se encuentra como cuarto elemento del arreglo
83          $orf[3] = reverse($orf[3]);
84
85          # Traduce cada nucleótido del cuarto elemento del arreglo a su nucleótido antisentido
86          # correspondiente
87          $orf[3] =~ tr/AUGC/UACG/;
88
89          # Copia la cadena del cuarto elemento del arreglo al quinto elemento del arreglo
90          $orf[4] = $orf[3];
91
92          # Elimina el primer caracter del quinto elemento del arreglo
93          $orf[4] =~ s/^.//;
94
95          # Copia la cadena del cuarto elemento del arreglo al sexto elemento del arreglo
96          $orf[5] = $orf[3];
97
98          # Elimina los dos primeros caracteres del sexto elemento del arreglo
99          $orf[5] =~ s/^.{2}//;
100
101         # Regresa los parámetros obtenidos por la función
102         return @orf;
103     }
104
105     # Subrutina     : sanitize_seq()
106     # Función       : Elimina de una secuencia todos los caracteres ajenos al tipo de secuencia
107     # Modo de uso   : $seq = sanitize_seq($seq, $seq_type)
108     # Parámetros    : $_[0] Una cadena de caracteres ($seq)
109     #                 $_[1] Una cadena con el tipo de secuencia que hay que validar (ej. "nucleotide")
110     #                       ($seq_type)
111     # Regresa       : $_[0] La cadena enviada como parámetro validada al tipo de secuencia solicitado o
112     #                       'undef' en caso de error ($seq)
113     sub sanitize_seq {
114
115         # Captura los parámetros enviados a la función
116         my ($seq, $seq_type) = @_;
117
118         # Elimina los saltos de línea de la secuencia enviada como parámetro
119         chomp $seq;
120
121         # Cambia la cadena a mayúsculas
122         $seq = uc($seq);
123
124         # Si el tipo de secuencia es "nucleotide"
125         if ($seq_type eq "nucleotide") {
126
127             # Elimina todos los caracteres que no son nucleótidos
128             $seq =~ s/[^AUGCT]//g;
129
130         # Si el tipo de secuencia es "aminoacid"
131         } elsif ($seq_type eq "aminoacid") {
132
133             # Elimina todos los caracteres que no son aminoácidos
```

```
134             $seq =~ s/[^ACDEFGHIKLMNPQRSTVWY\*]//g;
135         }
136 
137         # Revisa que la longitud de la secuencia validada sea mayor que cero o regresa 'undef'
138         length($seq) > 0 or return;
139 
140         # Regresa los parámetros obtenidos por la función
141         return $seq;
142     }
143 
144     # Subrutina     : split_by_stops()
145     # Función       : Obtiene subsecuencias separando en cada STOP(*) una proteína
146     # Modo de uso   : @sub_protein = split_by_stops($protein)
147     # Parámetros    : $_[0] Una cadena de proteína previamente validada por la función sanitize_seq() o
148     #                       generada por la función translate() ($protein)
149     # Regresa       : $_[0] Un arreglo con las subsecuencias obtenidas (@sub_protein)
150     sub split_by_stops {
151 
152         # Captura los parámetros enviados a la función
153         my ($protein) = @_;
154 
155         # Separa en cada STOP(*) la secuencia enviada como parámetro y regresa en un arreglo el
156         # resultado
157         return split(/\*/, $protein);
158     }
159 
160     # Subrutina     : translate()
161     # Función       : Traduce una secuencia de mRNA a proteína
162     # Modo de uso   : $protein = translate($seq, %trans_table)
163     # Parámetros    : $_[0] Una cadena de mRNA generada por la función DNA_2_mRNA() ($seq)
164     #                 $_[1] Un hash generado por la función get_trans_table() definida en CodonTable.pm
165     #                       (%trans_table)
166     # Regresa       : $_[0] Una cadena con la proteína obtenida ($protein)
167     sub translate {
168 
169         # Captura los parámetros enviados a la función
170         my ($seq, %trans_table) = @_;
171 
172         # Inicia la cadena donde se almacenarán los aminoácidos correspondientes a cada codón
173         my $protein = "";
174 
175         # Por cada codón (triplete) contenido dentro de la secuencia enviada como parámetro añade un
176         # salto de línea
177         $seq =~ s/(.{3})/$1\n/g;
178 
179         # Ejecuta un ciclo por cada triplete contenido en la secuencia
180         for my $triplet (split(/\n/, $seq)) {
181 
182             # Si el triplete existe en la tabla de traducción enviada como parámetro
183             if (exists $trans_table{$triplet}) {
184 
185                 # Concatena el aminoácido correspondiente al final de la cadena que se va a regresar
186                 $protein .= $trans_table{$triplet};
187             }
188         }
189 
190         # Regresa los parámetros obtenidos por la función
191         return $protein;
192     }
193 
194     1;
```

/home/prosa/templates/aminoacid.tmpl

```
1     <html>
2     <head>
3     <title>Resultados de ProSA::Protein Sequence Analyzer</title>
4     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5     <meta http-equiv="Content-Language" content="es">
6     <meta http-equiv="Expires" content="0">
7     <meta http-equiv="Pragma" content="no-cache">
8     <meta http-equiv="Reply-to" content="web-biol@campus.iztacala.unam.mx">
9     <meta name="Author" content="Mauricio Herrera Cuadra">
10    <meta name="Description" content="Prosa::Protein Sequence Analyzer">
11    <meta name="Copyright" content="Copyright © Mauricio Herrera Cuadra 2003">
12    <meta name="Keywords" content="prosa, protein, sequence, analyzer, protein sequence analyzer">
13    </head>
14    <body>
15    <h1>Resultados de ProSA::Protein Sequence Analyzer</h1>
16    <h2>La secuencia de prote&iacute;na introducida fu&eacute;:</h2>
17    <pre><tmpl_var name="sequence"></pre>
18    <hr width="100%">
19    <p>La duraci&oacute;n del an&aacute;lisis fu&eacute;: <b><tmpl_var name="elapsed_time"></b></p>
20    <hr width="100%">
21    <h2>La secuencia de prote&iacute;na utilizada para el an&aacute;lisis fu&eacute;:</h2>
22    <pre><tmpl_var name="protein"></pre>
23    <p>Esta secuencia contiene <b><tmpl_var name="sp_number"></b> subsecuencia(s):</p>
24    <tmpl_loop name="sub_protein">
25    <blockquote>
26      <h3>La subsecuencia:</h3>
27      <pre><tmpl_var name="sub_protein"></pre>
28      <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
      <b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
29      <tmpl_if name="PROSITE_search">
30      <tmpl_loop name="PROSITE_search">
31      <blockquote>
32        <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
      ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
      name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
33        <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
34        <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
35        <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
36      </blockquote>
37      </tmpl_loop>
38      </tmpl_if>
39    </blockquote>
40    </tmpl_loop>
41    <hr width="100%">
42    <p><a href="/prosa.html">Analizar otra secuencia</a></p>
43    </body>
44    </html>
```

/home/prosa/templates/error.tmpl

```
1     <html>
2     <head>
3     <title>Error - ProSA::Protein Sequence Analyzer</title>
4     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5     <meta http-equiv="Content-Language" content="es">
6     <meta http-equiv="Expires" content="0">
7     <meta http-equiv="Pragma" content="no-cache">
8     <meta http-equiv="Reply-to" content="web-biol@campus.iztacala.unam.mx">
9     <meta name="Author" content="Mauricio Herrera Cuadra">
10    <meta name="Description" content="Prosa::Protein Sequence Analyzer">
11    <meta name="Copyright" content="Copyright © Mauricio Herrera Cuadra 2003">
12    <meta name="Keywords" content="prosa, protein, sequence, analyzer, protein sequence analyzer">
13    </head>
14    <body>
15    <h1>Error - ProSA::Protein Sequence Analyzer</h1>
16    <p><b><tmpl_var name="error"></b></p>
17    <hr width="100%">
18    <p><a href="/prosa.html">Intentar nuevamente</a></p>
19    </body>
20    </html>
```

/home/prosa/templates/nucleotide.tmpl

```
1     <html>
2     <head>
3     <title>Resultados de ProSA::Protein Sequence Analyzer</title>
4     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5     <meta http-equiv="Content-Language" content="es">
6     <meta http-equiv="Expires" content="0">
7     <meta http-equiv="Pragma" content="no-cache">
8     <meta http-equiv="Reply-to" content="web-biol@campus.iztacala.unam.mx">
9     <meta name="Author" content="Mauricio Herrera Cuadra">
10    <meta name="Description" content="Prosa::Protein Sequence Analyzer">
11    <meta name="Copyright" content="Copyright © Mauricio Herrera Cuadra 2003">
12    <meta name="Keywords" content="prosa, protein, sequence, analyzer, protein sequence analyzer">
13    </head>
14    <body>
15    <h1>Resultados de ProSA::Protein Sequence Analyzer</h1>
16    <h2>La secuencia introducida fu&eacute;:</h2>
17    <pre><tmpl_var name="sequence"></pre>
18    <hr width="100%">
19    <p>La durac&oacute;n del an&aacute;lisis fu&eacute;: <b><tmpl_var name="elapsed_time"></b></p>
20    <hr width="100%">
21    <h2>La secuencia de mRNA utilizada para la traducci&oacute;n fu&eacute;:</h2>
22    <pre><tmpl_var name="mRNA_sequence"></pre>
23    <p>Contiene <b><tmpl_var name="mRNA_seq_length"></b> nucle&oacute;tidos y su peso molecular es:
      <b><tmpl_var name="mRNA_seq_mw"> Da</b>.</p>
24    <p>Esta compuesta por:<br>
25      <b><tmpl_var name="a_number"></b> Adeninas <b>(<tmpl_var name="a_percent"> %)</b>.<br>
26      <b><tmpl_var name="u_number"></b> Uracilos <b>(<tmpl_var name="u_percent"> %)</b>.<br>
27      <b><tmpl_var name="g_number"></b> Guaninas <b>(<tmpl_var name="g_percent"> %)</b>.<br>
28      <b><tmpl_var name="c_number"></b> Citosinas <b>(<tmpl_var name="c_percent"> %)</b>.</p>
29    <p>El porcentaje de AU es: <b><tmpl_var name="au_percent"> %</b> y el de GC es: <b><tmpl_var
      name="gc_percent"> %</b>.</p>
30    <hr width="100%">
31    <p>El <b>C&oacute;digo Gen&eacute;tico</b> seleccionado para la traducci&oacute;n fu&eacute;: <b><a
      href="http://www.ncbi.nlm.nih.gov/htbin-post/Taxonomy/wprintgc?mode=c#SG<tmpl_var name=table>"
      target="_blank"><tmpl_var name="table_id"></a></b>.</p>
32    <hr width="100%">
33    <h2>El 1er marco de lectura genera la secuencia:</h2>
34    <pre><tmpl_var name="protein_1"></pre>
35    <p>Esta secuencia contiene <b><tmpl_var name="sp1_number"></b> subsecuencia(s):</p>
36    <tmpl_loop name="sub_protein_1">
37    <blockquote>
38      <h3>La subsecuencia:</h3>
39      <pre><tmpl_var name="sub_protein"></pre>
40      <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
      <b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
41      <tmpl_if name="PROSITE_search">
42      <tmpl_loop name="PROSITE_search">
43      <blockquote>
44        <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
      ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
      name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
45        <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
46        <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
47        <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
48    </blockquote>
49    </tmpl_loop>
50    </tmpl_if>
51    </blockquote>
52    </tmpl_loop>
53    <hr width="100%">
54    <h2>El 2do marco de lectura genera la secuencia:</h2>
55    <pre><tmpl_var name="protein_2"></pre>
56    <p>Esta secuencia contiene <b><tmpl_var name="sp2_number"></b> subsecuencia(s):</p>
57    <tmpl_loop name="sub_protein_2">
```

```
<blockquote>
  <h3>La subsecuencia:</h3>
  <pre><tmpl_var name="sub_protein"></pre>
  <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
<b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
  <tmpl_if name="PROSITE_search">
  <tmpl_loop name="PROSITE_search">
  <blockquote>
    <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
    <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
    <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
    <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
  </blockquote>
  </tmpl_loop>
  </tmpl_if>
</blockquote>
</tmpl_loop>
<hr width="100%">
<h2>El 3er marco de lectura genera la secuencia:</h2>
<pre><tmpl_var name="protein_3"></pre>
<p>Esta secuencia contiene <b><tmpl_var name="sp3_number"></b> subsecuencia(s):</p>
<tmpl_loop name="sub_protein_3">
<blockquote>
  <h3>La subsecuencia:</h3>
  <pre><tmpl_var name="sub_protein"></pre>
  <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
<b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
  <tmpl_if name="PROSITE_search">
  <tmpl_loop name="PROSITE_search">
  <blockquote>
    <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
    <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
    <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
    <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
  </blockquote>
  </tmpl_loop>
  </tmpl_if>
</blockquote>
</tmpl_loop>
<hr width="100%">
<h2>El 4to marco de lectura genera la secuencia:</h2>
<pre><tmpl_var name="protein_4"></pre>
<p>Esta secuencia contiene <b><tmpl_var name="sp4_number"></b> subsecuencia(s):</p>
<tmpl_loop name="sub_protein_4">
<blockquote>
  <h3>La subsecuencia:</h3>
  <pre><tmpl_var name="sub_protein"></pre>
  <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
<b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
  <tmpl_if name="PROSITE_search">
  <tmpl_loop name="PROSITE_search">
  <blockquote>
    <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
    <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
    <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
    <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
  </blockquote>
  </tmpl_loop>
  </tmpl_if>
</blockquote>
</tmpl_loop>
```

```
<hr width="100%">
<h2>El 5to marco de lectura genera la secuencia:</h2>
<pre><tmpl_var name="protein_5"></pre>
<p>Esta secuencia contiene <b><tmpl_var name="sp5_number"></b> subsecuencia(s):</p>
<tmpl_loop name="sub_protein_5">
<blockquote>
  <h3>La subsecuencia:</h3>
  <pre><tmpl_var name="sub_protein"></pre>
  <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
<b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
  <tmpl_if name="PROSITE_search">
  <tmpl_loop name="PROSITE_search">
  <blockquote>
    <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
    <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
    <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
    <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
  </blockquote>
  </tmpl_loop>
  </tmpl_if>
</blockquote>
</tmpl_loop>
<hr width="100%">
<h2>El 6to marco de lectura genera la secuencia:</h2>
<pre><tmpl_var name="protein_6"></pre>
<p>Esta secuencia contiene <b><tmpl_var name="sp6_number"></b> subsecuencia(s):</p>
<tmpl_loop name="sub_protein_6">
<blockquote>
  <h3>La subsecuencia:</h3>
  <pre><tmpl_var name="sub_protein"></pre>
  <p>Contiene <b><tmpl_var name="aa_number"></b> amino&aacute;cidos y su peso molecular es:
<b><tmpl_var name="prot_seq_mw"> Da</b>.</p>
  <tmpl_if name="PROSITE_search">
  <tmpl_loop name="PROSITE_search">
  <blockquote>
    <p>Se encontr&oacute;: <b><a href="http://www.expasy.org/cgi-bin/prosite-search-
ful?SEARCH=<tmpl_var name=accession>" target="_blank"><tmpl_var name="description"> (<tmpl_var
name="id">)</a></b> en la posici&oacute;n: <b><tmpl_var name="position"></b>
    <br>El fragmento concordante fu&eacute;: <b><tmpl_var name="match"></b>
    <br>El patr&oacute;n de b&uacute;squeda fu&eacute;: <b><tmpl_var name="pattern"></b>
    <br>La expresi&oacute;n regular equivalente es: <b><tmpl_var name="regexp"></b></p>
  </blockquote>
  </tmpl_loop>
  </tmpl_if>
</blockquote>
</tmpl_loop>
<hr width="100%">
<p><a href="/prosa.html">Analizar otra secuencia</a></p>
</body>
</html>
```

Bibliografía

1. **Adeleke, A. A., Fields, B. S., Benson, R. F., Daneshvar, M. I., Pruckler, J. M., Ratcliff, R. M., Harrison, T. G., Weyant, R. S., Birtles, R. J., Raoult, D. y Halablab, M. A**. 2001. *Legionella drozanskii* sp. nov., *Legionella rowbothamii* sp. nov. and *Legionella fallonii* sp. nov.: three unusual new *Legionella* species. *International Journal of Systematic and Evolutive Microbiology*. 51, 1151-1160.
2. **Bedell, J., Korf, I. y Yandell, M**. 2003. *BLAST*. O'Reilly & Associates, Inc. 1ra Edición.
3. **Beppu, J**. 2002. 9 Power Tools Are Enough. Unix Tools that will Improve Your Life. *Linux Magazine*. *http://www.linux-mag.com/2002-04/unix_tools_01.html*
4. **Burger, G., Lang, B. F. y Gray, W. M. M. W**. 2000. Phylogenetic relationships of stramenopile algae, based on complete mitochondrial genome sequences. Sin publicar.
5. **Burger, G., Plante, I., Lonergan, K. M. y Gray, M. W**. 1995. The mitochondrial DNA of the amoeboid protozoon, *Acanthamoeba castellanii*: complete sequence, gene content and genome organization. *Journal of Molecular Biology*. 245, 522-537.
6. **Chen, H. y Sharp, B. M**. 2002. Oliz, a suite of Perl scripts that assist in the design of microarrays using 50mer oligonucleotides from the 3' untranslated region. *BMC Bioinformatics*. 3, 27-33.
7. **Chervitz, S. A., Fuellen, G., Dagdigian, C., Brenner, S. E., Birney, E. y Korf, I**. 1998. Bioperl: Standard Perl Modules for Bioinformatics. *Bits Journal*. *http://www.bitsjournal.com/bioperl.html*
8. **Christiansen, T. y Torkington, N**. 2003. *Perl Cookbook*. O'Reilly & Associates, Inc. 2da Edición.
9. **Elzanowski, A. y Ostell, J**. 2000. *The Genetic Codes*. National Center for Biotechnology Information (NCBI). *http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi?mode=t*
10. **Friedl, J**. 2002. *Mastering Regular Expressions*. O'Reilly & Associates, Inc. 2da Edición.
11. **Gibas C. y Jambeck P**. 2001. *Developing Bioinfomatics Computer Skills*. O'Reilly & Associates, Inc. 1ra Edición.
12. **Gilbert, D**. 2002. Pise: Software for building bioinformatics webs. *Briefings in Bioinformatics*. 3, 405-409.
13. **Guelich, S., Gundavaram, S. y Birznieks, G**. 2000. *CGI Programming with Perl*. O'Reilly & Associates, Inc. 2da Edición.
14. **Hermjakob, H., Fleischmann, W. y Apweiler, R**. 1999. Swissknife – 'lazy parsing' of SWISS-PROT entries. *Bioinformatics*. 15, 771-772.
15. **Krieger, E. y Vriend, G**. 2002. Models@Home: distributed computing in bioinformatics using a screensaver based approach. *Bioinformatics*. 18, 315–318.
16. **Kuck, U., Jekosch, K. y Holzamer, P**. 2000. DNA sequence analysis of the complete mitochondrial genome of the green alga *Scenedesmus obliquus*: evidence for UAG being a leucine and UCA being a non-sense codon. *Gene*. 253, 13-18.
17. **Kumazawa, Y. y Nishida, M**. 1999. Complete mitochondrial DNA sequences of the green turtle and blue-tailed mole skink: statistical evidence for archosaurian affinity of turtles. *Molecular Biology Evolution*. 16, 784-792.
18. **Laurie, B y Laurie, P**. 1999. *Apache: The Definitive Guide*. O'Reilly & Associates, Inc. 2da Edición.
19. **Lehey, G**. 2003. *The Complete FreeBSD*. O'Reilly & Associates, Inc. 4ta Edición.
20. **Leon, D. y Markel, S**. 2003. *Sequence Analysis in a Nutshell*. O'Reilly & Associates, Inc. 1ra Edición.
21. **Letondal, C**. 2001. A Web interface generator for molecular biology programs in Unix. *Bioinformatics*. 17, 73-82.
22. **Lewis, D. L., Farr, C. L. y Kaguni, L. S**. 1995. *Drosophila melanogaster* mitochondrial DNA: completion of the nucleotide sequence and evolutionary comparisons. *Insect Molecular Biology*. 4, 263-278.
23. **Lexa, M., Horak, J. y Brzobohaty, B**. 2001. Virtual PCR. *Bioinformatics*. 17, 192-193.

24. **Lim, A. y Zhang, L**. 1999. WebPHYLIP: a web interface to PHYLIP. *Bioinformatics*. 15, 1068-1069.
25. **Loewe, L**. 2002. Global computing for bioinformatics. *Briefings in Bioinformatics*. 3, 377-388.
26. **Mangalam, H**. 2002. The Bio* toolkits – a brief overview. *Briefings in Bioinformatics*. 3, 296-302.
27. **Mulder, N. J. y Apweiler, R**. 2001. Tools and resources for identifying protein families, domains and motifs. *Genome Biology*. 3, 1-8.
28. **Narasimhan, G., Bu, C., Gao, Y., Wang, X., Xu, N. y Mathee, K**. 2002. Mining Protein Sequences for Motifs. *Journal of Computational Biology*. 9, 707-720.
29. **Peek, J., O'Reilly, T. y Loukides, M**. 1997. *UNIX Power Tools*. O'Reilly & Associates, Inc. 2da Edición.
30. **Petropoulos, C. J**. 1997. Appendix 2: Retroviral taxonomy, protein structure, sequences, and genetic maps. Publicado en *RETROVIRUSES*. Pág. 757. Cold Spring Harbor Laboratory Press.
31. **Saccharomyces Genome Database**. 1999. Department of Genetics, Stanford University. *yeast-curator@genome.stanford.edu*
32. **Schwartz, R. y Phoenix, T**. 2001. *Learning Perl*. O'Reilly & Associates, Inc. 3ra Edición.
33. **Sigrist, C. J. A., Cerutti, L., Hulo, N., Gattiker, A., Falquet, L., Pagni, M., Bairoch, A. y Bucher, P**. 2002. PROSITE: A documented database using patterns and profiles as motif descriptors. *Briefings in Bioinformatics*. 3, 265-274.
34. **Spainhour, S., Siever, E. y Patwardhan, N**. 2002. *Perl in a Nutshell*. O'Reilly & Associates, Inc. 2da Edición.
35. **Srinivasan, S**. 1997. *Advanced Perl Programming*. O'Reilly & Associates, Inc. 1ra Edición.
36. **Stallman, R**. 1999. The GNU Operating System and the Free Software Movement. Publicado en *Open Sources: Voices from the Open Source Revolution*. O'Reilly & Associates, Inc. 1ra Edición.
37. **Stein, L**. 1996. How Perl Saved the Human Genome Project. *The Perl Journal*. 1, 5-9.
38. **Stein, L**. 2002. Creating a bioinformatics nation. *Nature*. 417, 119-120.
39. **Stubblebine, T**. 2003. *Regular Expression Pocket Reference*. O'Reilly & Associates, Inc. 1ra Edición.
40. **Takagi, T., Iwaasa, H., Yuasa, H., Shikama, K., Takemasa, T. y Watanabe, Y**. 1993. Primary structure of *Tetrahymena* hemoglobins. *Biochim. Biophys. Acta*. 1173, 75-78.
41. **Tan, M., Liang, A., Bruenen-Nieweler, C. y Heckmann, K**. 2001. Programmed translational frameshifting is likely required for expressions of genes encoding putative nuclear protein kinases of the ciliate *Euplotes octocarinatus*. *Journal of Eukaryotic Microbiology*. 48, 578-582.
42. **Tisdall, J**. 2001. *Beginning Perl for Bioinformatics*. O'Reilly & Associates, Inc. 1ra Edición.
43. **Tisdall, J**. 2001. Parsing Protein Domains with Perl. Publicado en *Perl.com*. O'Reilly & Associates, Inc. *http://www.perl.com/pub/a/2001/11/16/perlbio2.html*
44. **Tisdall, J**. 2003. *Mastering Perl for Bioinformatics*. O'Reilly & Associates, Inc. 1ra Edición.
45. **Vromans, J**. 2002. *Perl Pocket Reference*. O'Reilly & Associates, Inc. 4ta Edición.
46. **Wall, L., Christiansen, T. y Orwant, J**. 2000. *Programming Perl*. O'Reilly & Associates, Inc. 3ra Edición.
47. **Wong, C**. 1997. **Web Client Programming with Perl**. O'Reilly & Associates, Inc. 1ra Edición.
48. **Yuan, J., Amend, A., Borkowski, J., DeMarco, R., Bailey, W., Liu, Y., Xie, G. y Blevins, R**. 1999. MULTICLUSTAL: a systematic method for surveying Clustal W alignment parameters. *Bioinformatics*. 15, 862-863.

Página cero

Los libros suelen tener un principio y un fin. Este no, y tal página podría encontrarse en cualquier sitio y, por descontado, fuera del libro.

Mi vida continúa, no conozco su principio ni tampoco su conclusión.

En un ángulo de mi cuarto, pendiente de un tenue cordón, cuelga el "mundo". Podemos mirar a su través, ya que consta de dos simples anillos de paja. Algún recorte en forma de estrella recuerda la Navidad; pero esto apenas tiene importancia. Es más, una de las cuatro estrellas se ha caído con el transcurso de los años. La esfera, sin embargo, continúa girando lentamente, unas veces hacia un lado y otras hacia el otro. Raras veces está quieta.

He llamado a estos aros de paja "el mundo".

De cuando en cuando, mientras pasa el tiempo, me quedo mirando estos ceros de paja y pienso en ellos.

Porque un cero no es nada; pero un cero que ha comenzado a girar debe forzosamente ser algo…

Kurt Diemberger
"Entre cero y ocho mil metros"