



UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE CONTADURIA Y ADMINISTRACION

TUTORIAL "CELLA", UNA APLICACIÓN DE
ENSEÑANZA ASISTIDA POR COMPUTADORA

SEMINARIO DE INVESTIGACION INFORMATICA

QUE PARA OBTENER EL TITULO DE:
LICENCIADO EN INFORMATICA

PRESENTA:

MYRNA HERNANDEZ GUTIERREZ

ASESOR DEL SEMINARIO:

L.A. LUIS EDUARDO LOPEZ CASTRO



MEXICO, D.F.

1995

ACTUALIZADO A

1998

TESIS CON
FALLA DE ORIGEN

260355



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A TODOS LOS QUE DE UNA U OTRA FORMA HICIERON POSIBLE
QUE ESTE PROYECTO SE REALIZARA.

GRACIAS.

TUTORIAL CELLA ,

UNA APLICACIÓN DE ENSEÑANZA ASISTIDA POR COMPUTADORA.

INTRODUCCIÓN.....1

MARCO TEÓRICO.....3

CAPÍTULO I

ENSEÑANZA ASISTIDA POR COMPUTADORA (EAC).....7

1.1 ANTECEDENTES Y DESARROLLO.....8

1.2 ENSEÑANZA ASISTIDA POR COMPUTADORA Y MULTIMEDIA.....29

1.3 ENSEÑANZA ASISTIDA POR COMPUTADORA EN MÉXICO.....32

CAPÍTULO II

TIPOS DE EAC.....42

2.1 PROGRAMAS DE PRÁCTICA Y EJERCICIO.....44

2.2 LECCIONARIOS.....50

2.3 TUTORIALES.....52

2.4 SIMULADORES.....60

2.5 JUEGOS EDUCATIVOS.....62

CAPÍTULO III

ANÁLISIS, DISEÑO Y PRODUCCIÓN DEL TUTORIAL CELLA.....	69
3.1 ANÁLISIS DE NECESIDADES	71
3.2 DISEÑO INSTRUCCIONAL	74
3.3 DISPOSITIVOS DE USO.....	101
3.4 DESARROLLO DEL PROTOTIPO Y DEL PRODUCTO.....	121
3.5 EVALUACIÓN.....	173

CAPÍTULO IV

PROYECCIÓN A FUTURO

4.1 FUTURO DE LA EAC.....	180
4.2 PROYECCIÓN A FUTURO	182
CONCLUSIONES.....	185
BIBLIOGRAFÍA	188
APÉNDICES.....	191
A. 1 PROGRAMA CELULA.PAS	192
A. 2 LECCIÓN 2 DEL TUTORIAL CELLA.....	192
B. PANTALLAS DEL TUTORIAL CELLA.....	207
C. MANUAL DE INSTALACIÓN	213

INTRODUCCIÓN

La introducción de las computadoras en prácticamente todas las áreas a partir de los 80's, ha tenido un incremento importante en nuestro país. Las áreas impactadas en principio fueron la científica y la administrativa; más tarde alrededor del 85, fue cuando el área de enseñanza empezó a conocer las opciones que ésta tecnología le ofrecía.

Una de las características del hombre es dudar de lo que no comprende. Y en cada época, independientemente del contexto histórico y del grado de desarrollo alcanzado, las "innovaciones" han sido puestas en tela de juicio. Evidentemente la computación no constituye la excepción, pero con una variante: actualmente ya casi nadie duda de su eficacia ni cuestiona su potencial y alcance; ahora existe una búsqueda incesante de formas y áreas donde introducirla.

Con este enfoque el proceso educativo constituye un campo inmenso para la aplicación de la computación, ya que el proceso en enseñanza - aprendizaje no se reduce sólo a las relaciones directas entre educador y educando, sino al conjunto de acciones hacia el desarrollo integral del sujeto que aprende. De manera que es posible construir una alternativa educacional que impone un desafío y que implica romper algún esquema clásico.

Este hecho de "automatizar" la enseñanza, es la base del desarrollo de la Enseñanza Asistida por Computadora (EAC). Sin embargo la relativamente reciente

aparición en México de ésta rama trae como consecuencias poca aceptación por parte de los profesores, falta de software adecuado para el país, es decir, software producido para usuarios mexicanos. Además, en cuanto a los productores, el software de EAC ha sido criticado como deficiente en calidad e imaginación.

Si bien el objetivo principal de ésta tesis es crear un software educativo de tipo tutorial para alumnos de nivel medio básico del área de biología, específicamente del tema de la célula, con el fin de desarrollar una aplicación (PCE)¹ de acuerdo al programa COEEBA de la SEP; en una perspectiva más amplia, crear software que forme parte de la EAC desarrollada y producida en México. También dentro de los objetivos secundarios esta el de servir de apoyo a los productores y desarrolladores que busquen experiencias en la creación de software educativo.

Esta versión (CELLA 1.1) del proyecto se pretende, primero, incrementar el acervo del programa COEEBA y la difusión de la EAC dentro del país; después, servir como base a una segunda versión mejorada de CELLA, con mayores posibilidades técnicas en cuanto a desarrollo y uso; e incluso ser la base para desarrollos en niveles superiores (nivel medio superior).

"Así pues, mi propósito no es enseñar aquí el método que cada cual debe seguir para conducir bien su corazón, sino solamente mostrar de que manera he tratado yo de conducir el mío"

R. Descartes.

¹ Programas de Computación Educativos. Software educativo que desarrolló la SEP para utilizarlos en las escuelas públicas.

MARCO TEÓRICO

El progreso humano, la “civilización”, se fundamenta en el conocimiento, la educación es la forma de adquirir y transmitir éste conocimiento. México vive una etapa crítica en su historia junto con los demás países en vías de desarrollo de Latinoamérica y del mundo, sufre una crisis socioeconómica causada en gran parte por la falta de un adecuado desarrollo en las áreas de ciencia y tecnología.

De hecho, la “ciencia - arte” de la enseñanza no ha tenido un progreso importante en estos últimos tiempos. El factor humano sigue y seguirá siendo esencial en el proceso de transmisión de conocimientos y de integración social, implícitos en el planteamiento educativo convencional y tradicional.

La introducción de nuevas tecnologías y herramientas puestas a disposición de los educadores, tales como la informática, introduce problemas nuevos, aún sin resolver. La cuestión es estudiar la forma en que la informática puede ser asimilada por el sistema en forma productiva y creadora, no sólo pasiva.

La educación es una de las áreas más importantes desde un punto de vista estratégico en la modernización de México. Los cambios por los que pasa nuestro país, actualmente, exigen una reacción dentro del modelo educativo. Dichos cambios incluyen un crecimiento exponencial de la información disponible y una revolución informática y de las telecomunicaciones. Por lo tanto el alumno tiene que prepararse para manejar la tecnología y enfrentar otro tipo de cultura.

El alumno, tiene que aprender a aprender, a prepararse para una vida de aprendizaje; desde como tener acceso a la información que el necesita, trabajar en equipo, abordar problemas con un enfoque interdisciplinario y a cooperar con grupos o individuos de otras culturas. Esto exige, como propone Mary E. Meagher², que se redefina el concepto de aprendizaje como “un proceso social e integral, natural y divertido con énfasis en la solución de problemas tangibles”.

La educación plantea compromisos de crecimiento, aprendizaje y fomento a la creatividad. Para hacerlo, se requiere del uso de herramientas didácticas eficientes. Ahora la computadora se manifiesta como una de las más completas y dinámicas que apoyan el desarrollo de la capacidad creativa del ser humano. La computación es una de las áreas vitales que pueden impulsar cambios en el modelo educativo y al mismo tiempo que fortalecer la capacidad para desarrollar una ciencia y una tecnología propia.

El Dr. José Sarukhán³, afirma que la computación no está lejos de convertirse en un estricto sinónimo de educación. En lo que a educación se refiere es vital que la tecnología sea una parte integral de un nuevo modelo educativo que este de acuerdo y conduzca al tipo de cambios que exige el siglo XXI.

Los modelos educativos contemporáneos suponen la necesaria participación del educando en la construcción de sus conocimientos, así como la formación de hábitos, destrezas, actitudes y habilidades que le permitan la toma de conciencia,

² en “Una búsqueda de nuevos modelos educativos”

³ *ibidem*

con una actitud positiva y autovalorativa de la realidad. La abstracción de modelos a partir de situaciones problemáticas, en las que hay necesidad de manejar la flexibilidad del pensamiento, la estimación, la generación de estrategias, la superación de la frustración, la autoasignación de responsabilidades, etc.; conllevan el indispensable proceso de observación, análisis, comparación, conceptualización y formalización de las nociones y conceptos a aprender.

Como todo instrumento de carácter educativo la computadora requiere ser dirigida, la maquina por sí misma carece de la significancia real de sus potencialidades y es aquí donde se encuentra una de las grandes dificultades a vencer: la creación de auténticos entornos de carácter pedagógico que satisfagan adecuadamente las necesidades de quien aprende, y para esto, es indispensable formar de manera adecuada a los profesionales y productores para que sean capaces de formular verdaderos programas pedagógicos con los que el alumno pueda interactuar de una manera dinámica y significativa. Uno de los aspectos básicos para el cumplimiento de éste aspecto es recurrir directamente a quien por vocación se encuentra dentro de las acciones del fenómeno educativo, esto es, incluir a los educadores (expertos en contenido) en los equipos multidisciplinarios para la elaboración de software educativo.

La creación de programas educativos con el apoyo de la computadora presenta innumerables problemas, pues si bien es cierto que este equipo tiene infinitas posibilidades de desarrollo, también lo es que deben ser aprovechadas adecuadamente. La investigación dentro de el área de EAC trata de superar sus

mismos problemas y ya son innumerables los países, entre ellos el nuestro, en los que se han establecido proyectos educativos de ésta naturaleza.

El impresionante aumento de la población escolar ha afectado sensiblemente la relación maestro - alumno , y con ello la posibilidad de mayores logros en el aprendizaje, es aquí donde la computadora empieza a jugar un papel muy importante en el contexto educativo. Algunos escépticos hacen cuestionamientos importantes como: ¿Cómo disponer del suficiente equipo para atender adecuadamente las necesidades de una población escolar progresivamente creciente, cuando no se dispone de recursos financieros suficientes?, ¿Para que efectuar grandes inversiones cuando no se cuenta aún con software educativo conveniente?, ¿Se está pretendiendo sustituir al maestro con la computadora?. En ésta tesis se tratará de dar respuestas convincentes a favor de la EAC, tomando en cuenta que la incursión de la computadora en el área educativa representa una opción para revolucionarla, siempre y cuando la incursión de esta nueva tecnología sea manejada con inteligencia. De lo anterior dependerá su éxito, en este terreno como en otros.

CAPÍTULO I

ENSEÑANZA ASISTIDA POR COMPUTADORA (EAC)

¿QUE ES LA EAC?

Desde la aparición de la informática personal se consideró que las computadoras se convertirían en una poderosa herramienta para complementar el trabajo de los profesores.

El objetivo de la EAC no es el de sustituir a los maestros sino de ofrecer a los alumnos un medio eficiente para asimilar de manera personalizada y amena los conocimientos teóricos aprendidos en clase.

En años recientes la Enseñanza Asistida por Computadora (EAC) ha ganado aceptación en centros educacionales que van desde el jardín de niños hasta las universidades. El uso de software educativo se está incrementando en las escuelas públicas y privadas, universidades, programas de entrenamiento empresarial y militar e incluso hasta en los hogares. Sin embargo el software de EAC ha sido el blanco de algunas críticas. Defectos que no serían aceptados en otros medios educacionales tales como libros de texto y audiovisuales, pueden todavía ser encontrados en el software educativo. Entre los problemas más frecuentemente mencionados están el de poca documentación escrita⁴, programas que no utilizan la

⁴ además ésta información es pobre.

capacidad de la computadora eficientemente, diseño pedagógico pobre, el contenido presenta errores u omisiones, retroalimentación (*feedback*) inadecuada, falta de control del usuario y programas poco imaginativos y simples. Estos problemas pueden ser debidos a la relativa novedad de la EAC y su rápida introducción en ambientes educativos. La mayoría de los productores y usuarios de software educativo tienen relativamente poca experiencia en el ambiente de la EAC.

Hay un enorme potencial para la EAC, pero para explotarlo los productores deben aprender como escribir mejores programas y los maestros como usarlos eficaz y eficientemente. Los buenos programas la gran mayoría de las veces son resultado de una adecuada planeación por parte de aquellos que entienden el uso de la computadora como un medio para impartir enseñanza.

1.1 ANTECEDENTES Y DESARROLLO

La historia de la EAC es breve comparada con otros tipos de herramientas educativas tales como libros, audiovisuales, etc. "[...] Cabe destacar que las primeras utilizaciones del ordenador en el campo educativo tienen ya más de veinticinco años, lo que pone en evidencia la temprana visión de los investigadores en torno a las aportaciones que estas máquinas podrían conferir en éste terreno."⁵

⁵ Gros, B. . "Aprender mediante el ordenador". p.21

LOS PRIMEROS ESFUERZOS EN ENSEÑANZA PROGRAMADA.

Muchos de los principios básicos de la EAC fueron establecidos antes de que las computadoras fueran utilizadas para la enseñanza. En la década de los cincuentas los programadores instruccionales comenzaron a buscar como resolver problemas educativos aplicando las técnicas del análisis de comportamiento a través de instrucción programada. Muchas autoridades marcaron este como el inicio de este movimiento en 1954 con la publicación del artículo del Dr. B. F. Skinner "The science of learning and teaching" (Markle, 1964). A diferencia de otros psicólogos de la época, Skinner vio un fuerte paralelismo entre sus actividades en el laboratorio acerca del comportamiento animal y practicas que podían mejorar la educación. Skinner definió acertadamente el aprendizaje como "un cambio en el comportamiento". El creía que los principios de estímulo y recompensa caracterizados por una respuesta inmediata podían dar enseñanza. De esta teoría básica la temprana teoría de la instrucción programada (IP), la cual era en su mayoría en forma de texto, surgió. La instrucción programada se divide en bloques o unidades los cuales a su vez se dividen en breves instrucciones secuenciales (*frames*). El estudiante lee las instrucciones secuenciales y responde las preguntas formuladas frecuentemente. El estudiante es inmediatamente provisto de la respuesta correcta. Si responde incorrectamente se le da la dirección para repetir el segmento apropiado de la lección. La instrucción es entonces individualizada, automatizada y caracterizada por un reforzamiento inmediato y un involucramiento activo del estudiante en el proceso de aprendizaje. Un alto grado de conocimiento de la materia es virtualmente asegurado para el estudiante que completa la secuencia. Así como es muy difícil avanzar sin haber aprendido las secuencias anteriores.

La instrucción programada fue rápidamente aplicada a las "maquinas de enseñar" las cuales aparecieron al final de la década de los años cincuentas y principios de los sesentas. Algunos de estos dispositivos guardaban el registro de aprovechamiento del estudiante, con cronómetros incorporados, o incluso entregaban dulces u otras recompensas por buen aprovechamiento. Las "maquinas de enseñar" y la Instrucción Programada fueron utilizadas por escuelas en distintos niveles, servicios militares y programas de autoaprendizaje a lo largo de la década de los sesentas. Cursos enteros fueron "programados" en forma de texto. El más popular fue probablemente el "*english 3200*" que era un curso de gramática para ingles básico en forma de texto. El inicio, desarrollo y terminación del "*english 3200*" fue aceptado por muchas universidades como una alternativa a los cursos de ingles tradicional a través de un profesor.

La técnica de Instrucción Programada fue redefinida con la experiencia. Los primeros programas eran lineales, o sea que cada estudiante avanzaba a lo largo de una sola trayectoria a través de una secuencia de instrucciones fijas. En cambio en tanto la Instrucción Programada fue evolucionando la técnica de "ramificación" se empezó a utilizar para permitir a los estudiantes no tener que avanzar por nodos que ya conocían u obtener información adicional en áreas que se les dificultaban. Los programas empiezan a incluir pre - test y post - test, sumarios, resúmenes, test de diagnostico e incluso opciones para el estudiante. Uno de los pioneros en el investigación de ramificaciones para Instrucción Programada fue Norman Crowder quien produjo un trabajo titulado "*Intrinsic programming*". Crowder incorporo preguntas de opción múltiple en la Instrucción Programada (IP) las cuales fueron

usadas para determinar el material que el estudiante debería de ver. En los programas de Crowder cada respuesta a una pregunta de opción múltiple mandaba al estudiante a una secuencia (*frame*) diferente. Las preguntas de Crowder anticipaban errores aceptables que el estudiante podía cometer y construían la secuencia de retroalimentación adecuada para explicarle porqué la respuesta era incorrecta.

Susan Markle incorporó sus ideas con las de Skinner y Crowder creando una referencia estándar⁶ de la Instrucción Programada. No existe duda acerca de que la instrucción Programada trabajó y estuvo basada en principios de aprendizaje por sonidos, pero nunca logró un alto grado de popularidad. Por toda su sonoridad como una técnica educacional, la IP tuvo una gran caída. La mayoría de los estudiantes encontró tediosa y árida la IP y en forma generalizada no encajaba bien en los grupos con clases de horario fijo. Las críticas también apuntaron el hecho de que solo era efectiva enseñando materias de información de hechos y era de poco valor en aquellas de actitud y valores. Después de éste primer contacto con la técnica, los educadores le dieron la espalda considerando que era tan solo otra novedad pasajera en el ámbito educacional.

Estos primeros esfuerzos con IP mostraron que la programación de cualquier "conocimiento" es un proceso complejo que involucra muchos talentos. Markle en 1964 sumalizó estas habilidades y talentos como sigue:

⁶ En el libro "good frames and bad" (1964)

1. *Amplio conocimiento de la materia. Es requerido para formular los objetivos del programa, estructurando el orden en el cual los conceptos y habilidades van a ser introducidos, generando ejemplos, relacionando segmentos de actividades, y determinando la verdad o falsedad de enunciados hechos acerca del tema. Si el programador no tiene los conocimientos suficientes, debe trabajar muy estrechamente con alguien más que si los tenga.*
2. *Habilidades de comunicación. No existe sustituto para una correcta escritura o para el uso artístico de otros medios (en éste contexto artísticamente y habilidades de comunicación significan la precisión y claridad necesarias para llegar al estudiante, no el diseño artístico de apreciación).*
3. *Buen análisis del comportamiento en el estudiante. La amplia gama de técnicas disponibles permiten a cada programador un amplio rango de opciones.*
4. *Habilidades de diagnóstico. Esto se refiere a las habilidades en observación de los procesos de aprendizaje de los estudiantes individuales quienes constituyen los primeros "conejillos de indias" en el desarrollo de procesos de prueba.*
5. *Perfeccionamiento del modelo experimental en el proceso de validación final. Un conocimiento del diseño experimental de sonido, principios de construcción de test, y procedimientos muestra son útiles para cualquiera que intente mostrar el rango de aplicaciones de un conjunto de procedimientos y materiales educativos.*

McMeen cree que en muchas formas EAC puede ser vista como un legado de la Instrucción Programada. Es importante ver estos puntos de convergencia entre la IP y la EAC para entender más claramente lo que constituye buenas técnicas de programación. Por ejemplo el aprendizaje individualizado, auto instrucción,

conocimiento inmediato de la veracidad de la respuesta, y reforzamiento son tan importantes hoy como lo fueron para la IP hace más de treinta años.

PRIMEROS ESPFUERZOS EN LA EAC: 1955 - 1965.

Algunos de los primeros esfuerzos para usar las computadoras en la enseñanza tuvieron lugar a finales de la época de los años cincuentas y principios de los sesentas. En esta época las computadoras eran muy grandes y muy costosas. El inesperado lanzamiento del satélite soviético *Sputnik* en 1957 tuvo la mayor influencia en la educación en Estados Unidos, al mismo tiempo que las computadoras tuvieron su primera aplicación en el ambiente educativo. El impacto en los estadounidenses por el logro de los rusos se convirtió en un compromiso por la calidad de la educación en su país, especialmente en matemáticas y ciencia. Mientras la mayoría de los esfuerzos desarrollados por estas disciplinas ignoraban la computación por sus elevados costos, un proyecto en la Universidad de Stanford implicaba definir una mejor dirección para los materiales de enseñanza basados en computadora, de aquellas habilidades desarrolladas a través de practica - ejercicio EAC.

CORPORACIÓN DE CURRÍCULA COMPUTACIONAL.

El Dr. Patrick Suppes lideró un equipo que creó un sistema completo para computación basado en habilidades aritméticas prácticas. Este mismo equipo más tarde desarrolló ciertas aplicaciones para poder enseñar artes del lenguaje

elemental. Ambas están actualmente disponibles - en versión perfeccionada - para una minicomputadora de la Corporación de Curricula Computacional, compañía fundada por Suppes. Una considerable aportación del Dr. Suppes fue la demostración validada de que las computadoras pueden ser "profesores" efectivos. Suppes creía que el éxito de usar computadoras para enseñar requiere la creación de un gran número de programas estructurados más que lecciones de tópicos aislados. Algunos productores de software educativo todavía no han aceptado o reconocido la necesidad de planear e integrar cuidadosamente material educativo como textos, audiovisuales y equipo de laboratorio en lo que respecta a EAC.

PROYECTO PLATO.

En la Universidad de Illinois, otro proyecto tomó un diferente acercamiento para llegar a lo que es EAC. Mientras que la mayoría de los educadores aceptaban el equipo de cómputo existente como un hecho, el proyecto PLATO estableció una técnica de diseño de nuevo equipo de cómputo tanto de hardware como de software para la tarea específica de la enseñanza en una amplia variedad de áreas y para una mayor cantidad de estudiantes en forma simultánea. De esta forma una mayor cantidad de áreas escolares tan diversas como elementales llámese matemáticas, chino, ciencias veterinarias y química, fueron desarrolladas en este proyecto PLATO de la década de los sesentas. Los investigadores del proyecto PLATO fueron los pioneros en el uso de gráficas de color, pantallas sensibles al tacto (*touch screen*) y una amplia variedad de modos de enseñanza simulada y de interacción usuario - computadora de alto nivel, así como, por supuesto, el control en manos del usuario.

A pesar de los fondos federales y la compartición de costos entre los usuarios que podían utilizar el proyecto PLATO, el desarrollo e impulso se vio limitado porque los costos asociados con el sistema eran demasiado elevados como para lograr una incursión a gran escala en el medio educativo.

A través de los años los proyectos de innovación de hardware y software en general, han evolucionado y han servido como modelo para los productores de EAC. Los productos dados por el proyecto PLATO que en la actualidad se encuentran bajo una marca registrada de la Corporación Control Data, se encuentran en auge por todos los Estados Unidos y en algunos otros países, tanto en equipos de minicomputadora como de microcomputadora.

LOS COMIENZOS DE EAC EN LAS ESCUELAS PÚBLICAS DE USA.

Los primeros impactos de EAC en escuelas públicas a niveles primaria y secundaria fue gradual. Las aplicaciones eran en función de los recursos locales en ambos sentidos tanto humanos como de técnicos. Unos cuantos profesores fueron capacitados para tener acceso a los equipos de cómputo de universidades o empresas aledañas. La mayoría de los esfuerzos se concentraron en la enseñanza de programación y de algunos programas educativos (en código) hechos por estudiantes.

Estas tempranas experiencias demandaron que los programas pudieran ser primero convertidos a un medio legible por la computadora usando tarjetas perforadas. Los

paquetes de tarjetas tenían que ser tomados del centro de cómputo en el cual eran corridos como programas cuando había tiempo disponible. Los resultados se imprimían y analizaban más tarde. Los maestros que hacían esta pionera labor debían mostrar una paciencia y dedicación fuera de lo común, por lo tanto la oportunidad para los estudiantes de interactuar con computadoras fue gracias a la dedicación individual de algunos profesores. De alguna manera esto sigue vigente hasta la fecha y en mayor grado en nuestro país .

Mientras que hubo una dramática reducción en el costo, la complejidad y el tamaño de las computadoras, se dio también un dramático incremento en la potencialidad, facilidad de uso y disponibilidad de software. Sin embargo es todavía inalcanzable el hecho de que los profesores aprendan a manejar eficientemente el software incrementando el provecho de esta poderosa herramienta.

Un movimiento para personalizar la enseñanza comenzó en 1960 y continua a la fecha, (en Estados Unidos mas que en nuestro país y debido, por supuesto, a la "reciente" introducción de la computadora en México) esto se dio cuando estudiantes y padres de familia demandaron a las escuelas buscar la implementación de métodos enfocados a las necesidades únicas de cada estudiante. Los planes de educación personalizada vinieron a concebirse como pruebas de consenso académico para habilidades e intereses, y entonces los profesores hicieron esfuerzos para diferenciar el tipo de capacitación. Este movimiento no se origino con profesores de EAC, pero ciertamente promovió en la EAC que los profesores reconocieran la utilidad de la computadora como una herramienta para personalizar la educación.

LA EAC SURGE ENTRE 1965 Y 1970.

El periodo entre 1965 y 1970 mostró un cambio gradual en el medio de la educación computarizada. Los profesores y productores de software realizaron el inmenso esfuerzo requerido para producir un programa tipo EAC; pues aunque EAC no desapareció si perdió terreno frente a otras aplicaciones de cómputo en el ambiente educativo ya que los estudiantes comenzaron a usar la computadora como una simple herramienta de solución de problemas, y los profesores intentaron usar la computadora como una ayuda de instrucción más que como una máquina de enseñanza por si misma. La aplicación conocida como IMC (Instrucción Manejada por Computadora) surgió. Llegaron entonces los test por medio de computadora así como algunas otras tareas mecánicas y de grabación repetitiva asociadas con la educación.

Actualmente los buenos programas de EAC incluyen aspectos de IMC tales como almacenamiento de diagnóstico y prescripción y almacenamiento de test. Para 1970, cerca del 31% de las escuelas secundarias de los Estados Unidos estaban haciendo uso administrativo de las computadoras, mientras que solo un 13% las utilizaba en enseñanza. Sin embargo debe de ser aclarado que al decir 13% de las escuelas usaba computación en la enseñanza, no significa que el 13% de los profesores o el 13% de los estudiantes así lo hiciera. Los datos de Bukoski y Korotkin en 1976 mostraron que, en 1975, la media de estudiantes usando computadoras en escuelas era aproximadamente de 1,350 mientras que la media de terminales disponibles era de dos. En las clases de matemáticas y ciencias de la

computación se requerían casi todas para su uso. Por lo tanto el impacto de EAC hasta este punto estaba severamente limitado.

COMIENZA LA REVOLUCIÓN DE LA COMPUTADORA: ENTRE 1970-1980.

En la década de los setentas ocurrieron tres grandes cambios en la computación educativa. El primero fue el desarrollo y mercadotecnia de la minicomputadora. Los reducidos costos asociados con las minicomputadoras permitieron a algunas universidades y escuelas públicas comprar sistemas que pudieran ser compartidos para propósitos de enseñanza. Sin embargo las mayores aplicaciones de enseñanza de las minicomputadoras era para la enseñanza de programación y para IMC.

El segundo cambio mayor fue la invención de la pequeña, no muy cara y relativamente potente microcomputadora. Las primeras micros solamente se encontraban disponibles en forma de kit, pero la Sol20, Altos, Commodore Pet, Apple II, y Radio Shacks Model I surgieron como las primeras computadoras de aplicación a usuario. La tecnología hizo a las escuelas un ofrecimiento que no pudieron rechazar, la comunidad escolar acepto: la revolución computacional comenzó en este momento. Los usuarios de la época que detectaron el gran potencial de las computadoras en la educación presionaron para su aplicación en las escuelas. Algunos pensaron que estas podían hacer el aprendizaje divertido y sencillo para todos, mientras aseguraban altos grados de aprovechamiento académico. Sin embargo como todo proyecto sin fundamentos teóricos, estas predicciones fueron rápidamente desechadas.

El tercer mayor cambio en la computación educativa fue el incremento en la cantidad y variedad de software educativo. Para el final de la década de los setentas, la venta de lo que a veces es llamado "programas educativos de la primera generación" había alcanzado proporciones mayores y atrajo la atención de los gigantes publicadores. Algunos de estos productos de EAC eran pobremente producidos. La demanda por software educativo temporalmente excedió la producción y entonces se dio la producción rápida de programas (esta producción rápida trajo como consecuencia programas carentes de creatividad y frecuentemente conteniendo fallas técnicas) que saturaron el mercado.

Otros esfuerzos de EAC tomaron diferentes caminos. Dos grandes proyectos de EAC durante este periodo fueron el proyecto de la Televisión Informativa Controlada por Computo Interactivo y de Tiempo Compartido (TICCITC) y el nacimiento del Consorcio Computacional Educativo de Minnesota (CCEM). En el proyecto TICCITC un gran numero de estudiantes de comunidades universitarias fueron capacitados en matemáticas e ingles usando EAC y los resultados fueron cuidadosamente analizados. Los resultados indicaban mejoras significativas en el aprovechamiento y en la actitud de los estudiantes que usaron EAC. El proyecto CCEM fue el primero a gran escala para producir software de enseñanza de alta calidad, el cual era sistemáticamente desarrollado y probado. La documentación para programas de CCEM era frecuentemente extensa, incluyendo aquellos puntos como planes de impresión de lección, guías de estudio reproducibles y ejercicios prácticos. El software de CCEM fue bien recibido y prontamente aprendido lo cual le dio reputación de calidad. CCEM también fue pionero en duplicarse legalmente para las escuelas (licencia de uso) y empaquetamiento de múltiples copias para paquetes

de clases. El CCEM permanecía como un software educacional líder en publicación hasta principios de ésta década en Estados Unidos.

Los esfuerzos realizados por estos dos proyectos, el TICCIT y el CCEM, ayudaron a establecer el valor práctico de la EAC tanto para las instituciones educativas como para los usuarios.

En cuanto surgieron las computadoras personales (PC) los usuarios empezaron a copiar programas haciendo hasta cientos de copias, para compartirlos con otros estudiantes y maestros. Para combatir la "piratería de software" los productores utilizaron esquemas de protección en los mismos discos a manera de dificultar el copiado y proteger los derechos de autor los cuales fueron impresos y puestos en las etiquetas de los discos.

LA REVOLUCIÓN DE LAS COMPUTADORAS TOMA IMPORTANCIA (80'S).

La creciente demanda de microcomputadoras en escuelas y oficinas, creada por el gran desarrollo de hardware tan solo unos años antes, en 1970 dio un cambio radical con el uso de terminales teletipo enlazadas por medio de líneas telefónicas a grandes centros de cómputo. Para 1980 estas mismas escuelas, en Estados Unidos, compraban docenas de microcomputadoras con características tales como gráficos en color, juegos de caracteres para diferentes lenguajes y capacidades de generación de música. Las computadoras entonces no eran simplemente aceptadas por profesores y educadores, sino que definitivamente las preferían.

Así mismo con los equipos adquiridos, las escuelas se interesaban y compraban las promociones de software educativo, fue entonces cuando se dieron los primeros indicios de que las computadoras podrían remplazar a los profesores desplazando el elemento humano gradualmente de la educación y concebir por tanto la idea de que las computadoras fueran una manera efectiva y practica de llevar a cabo la enseñanza.

LA COMPUTADORA IMPERSONAL.

Muchos de los diferentes productos y modelos que se estaban dando en el mercado a principios de los ochentas fueron llamados Computadoras Personales, tales como la Commodore 64 y 128, la Radio Shak modelo III y modelo IV, la Texas Instruments modelo 99/4, eran algunos de los más populares, sin embargo la línea de computadoras Apple II (la II+, la IIe y la IIC) se convirtieron en las más populares en las escuelas públicas de Estados Unidos. También había computadoras Acorns y computadoras Atari y algunos otros tipos, pero la frustración de los usuarios residía en la incompatibilidad de software. Esta variedad de computadoras incompatibles causo un dilema para los productores de software educativo los cuales se vieron forzados a elegir ya fuera una marca exclusiva de cómputo o gastar una gran cantidad en tiempo y dinero desarrollando diferentes versiones de un mismo programa ajustado a cada una de las variedades existentes (o por lo menos a la mayoría).

IBM introdujo al mercado casi al final de 1981 la computadora personal (IBM PC), la cual estaba basada en un sistema operativo desarrollado por la Corporación Microsoft llamado MS - DOS. Posterior a esto la PC fue seguida por modelos más potentes como fueron el AT y XT. Una gran variedad de clones compatibles con IBM y basados en MS-DOS fueron introducidos, lo que origino que en solo algunos años las computadoras "MS-DOS" se habrían de convertir en el estándar computacional en el mundo de los negocios.

Para cuando IBM introdujo su PC, las escuelas públicas habían crecido en un ambiente Apple II. Muchas inclusive habían amasado una enorme cantidad de software dado que el agresivo mercado de Apple ofrecía soporte a la comunidad escolar combinado con el continuo mejoramiento de los nuevos productos de Apple II lo que permitía a la Corporación Apple mantener el liderazgo en ventas de equipo de cómputo al sector de educación pública. Las computadoras que utilizaban MS-DOS encontraban su mayor aplicación en centros educativos de alto nivel como universidades y clases de computación. Sin embargo la línea Apple II continuaba dominando las escuelas especialmente en aplicaciones de EAC, mientras que las computadoras que utilizaban MS-DOS dominaban en el mundo de los negocios.

EVALUACIÓN SISTEMÁTICA DE SOFTWARE .

Al tiempo que los educadores se sofisticaban como consumidores de software educativo fueron demandando a los productores mejores programas. Muchos modelos de evaluación de software fueron publicados y entonces los educadores

contaban con mejores herramientas de evaluación para los paquetes de software disponibles en el mercado. Una gran variedad de organizaciones comenzó a evaluar software educativo y publicar sus recomendaciones. La Asociación Nacional Educativa (NEA), la sociedad de Intercambio de Productos de Información Educativa (EPIE), y los laboratorios educativos regionales del noreste de Estados Unidos con el proyecto "Microsift" fueron los primeros en evaluar software a gran escala y con bases sistemáticas. Revistas educativas y publicaciones comenzaron a mostrar revisiones de software y varios libros de software recomendado para estos fines. Varias agencias de educación estatal, incluyendo a aquellas de Texas, California, Minnesota, etc., comenzaron a evaluar y recomendar software educativo que las escuelas podían comprar. La competencia entre vendedores y el aumento en la demanda de software de calidad por los consumidores dio lugar a una mejora significativa en el software educativo, la cual fue denominada "segunda generación de software".

CRONOLOGÍA DE EVENTOS.

En 1980 el Consejo Nacional de Maestros de Matemáticas (E.U.) emitió un itinerario de acciones a tomar: que eran básicamente recomendaciones para escuelas de matemáticas. Este trabajo fue la culminación de años de estudio incluyendo estudios de campo a nivel nacional tanto para educadores como para alumnos. El estudio recomendó varios cambios en este tipo de educación, entre los cuales se encuentran los siguientes:

1. La enseñanza de teoría de computación debería de ser incluida en la educación de cada estudiante.
2. El desarrollo de software de instrucciones de alta calidad debería ser responsabilidad conjunta entre publicadores y educadores.
3. La computación debería de utilizarse en aplicaciones tales como la solución de problemas, tutoriales, simuladores, servicios de soporte a maestros, y también en la enseñanza de programación.

El interés de los educadores en las computadoras como herramientas de enseñanza los motivo a expandir y desarrollar materiales de enseñanza computarizada. Mientras que un estudio en 1980 encontró que solo siete de los catorce publicadores⁷ principales de libros de texto, dedicaban mayor tiempo y recursos al desarrollo de software; para 1985 prácticamente todos lo hacían. Algunas pseudo empresas fundadas en 1970 se convirtieron en publicadores (vendedores) mayores de sus propios productos llegando a ser verdaderas pequeñas empresas. Fue entonces cuando algunos sectores educativos se dieron a la tarea de trabajar en sus propios desarrollos de software educativo y algunos profesores que ya tenían habilidades de programación produjeron software para sus propias materias.

Para mediados de la década de los ochentas la producción de software de cómputo educativo se había convertido en un gran negocio y en 1984 el número de productores de software educativo llegó a los quinientos.

⁷de publisher: relacionado con la publicación de temas de software o del software mismo.

AVANCES EN HARDWARE.

Durante éste periodo se desarrollaron nuevos productos de cómputo que proporcionaron nuevas posibilidades en el uso de las computadoras como herramienta para la enseñanza. Las computadoras Apple introdujeron el concepto Macintosh en 1984 que popularizaba el uso del *mouse* y el fácil uso para el usuario por ser orientado a gráficos. La Apple IIGS y la IBM System/2 las cuales incorporaban algunas de las características popularizadas por Macintosh aparecieron en el mercado para 1987. Cada nuevo producto era caracterizado por un incremento en la capacidad de memoria tanto en la computadora como en los dispositivos de almacenamiento externo, haciendo el desarrollo lo mas sofisticado posible para el programa de software. Los dispositivos y las técnicas podían aprovechar las capacidades de la amplia base instalada de Apple II y computadoras como IBM PC y sus clones, las cuales se encontraban ya instaladas en las escuelas públicas de Estados Unidos, haciéndolas más poderosas. Más tarde al final de la década de los ochentas Macintosh e IBM PS/DOS ganaron puntos en el mercado educativo.

La diversidad de medios (multimedia) como animación, gráficos en color, y voz sintetizada de buena calidad así como música, comenzaron a ser incorporados en los programas de EAC en esta década. También comenzaron a surgir programas con muchas trayectorias y varios niveles de complejidad. Algunos indicadores (revistas: "*Only the best*" y "*Electronic Learning*") de la tendencia de hardware y software arrojaron en noviembre de 1989, que se dio una reducción del 92% al 82%, con respecto a la edición del año anterior, en los programas escritos en Apple II. El 35%

de los programas listados en esta edición corrían en IBM PC's y compatibles, 40% abajo que en el año anterior, y 10% corrían en Macintosh, 5.7% arriba que el año anterior. Debe ser notorio que muchos programas de EAC son producidos en versiones para dos o más plataformas. Aunque estas estadísticas no proporcionan datos precisos del número y tipos de programas para EAC en uso, sí proporcionan alguna indicación de que tipos de computadora están siendo utilizados para aplicaciones de EAC en foros educativos.

Unos cuantos programas fueron incorporados en novedosos dispositivos auxiliares tales como vídeo discos láser, dispositivos de almacenamiento de datos en CD - ROM y sintetizadores de sonido, para aplicaciones de EAC, sin embargo estas innovaciones no fueron ampliamente implementadas, los educadores aceptaron éstas mejoras, pero preferían esperar a que estas bajaran de precio y aseguraran compatibilidad de marcas. Los productores por otro lado no querían arriesgar una inversión para software de EAC utilizando estos nuevos dispositivos hasta que el mercado fuera seguro. Nuevamente la falta de software inhibió el mayor desarrollo y uso de la tecnología de EAC.

ESTADO DE LA EAC HASTA LOS 90's.

Entre 1980 y 1990 las escuelas públicas de Estados Unidos adquirieron aproximadamente 2.5 millones de computadoras así como también una gran variedad de software y herramientas educativas. En 1981 más de una escuela de cada cinco tenía al menos una computadora, pero para 1989 casi todas las

escuelas tenían computadora. Para 1988 las escuelas públicas en los Estados Unidos promediaban una computadora para cada 30 estudiantes. La EAC está todavía en las etapas de desarrollo e implantación. Algunos educadores y "capacitadores" han decidido después de su primera interacción con la EAC, que las computadoras realmente no pueden enseñar. Este negativismo se basa en el hecho de que la EAC no ha superado las grandes expectativas de algunos precursores. La mayoría de los educadores sin embargo se han convencido de que el papel de las computadoras como maestros es válido y de hecho excitante y están constantemente en la búsqueda de nuevos y mejores programas educativos.

Desafortunadamente los países pioneros son los que cuentan ahora con una gran variedad de productores y desarrolladores, grandes y pequeños que producen programas de EAC en una amplia variedad de modos y aplicaciones.

UBICACIÓN DE LAS COMPUTADORAS.

En Estados Unidos, las escuelas están comprando un gran número de computadoras así como de programas educativos. El crecimiento mayor para escuelas públicas con aplicaciones de EAC en este país parece darse en las escuelas primarias. Una vez que las escuelas adquieren estos equipos de cómputo, uno de los puntos de discusión con los cuales se ven forzados a tratar es en donde deben quedar ubicadas las computadoras. Algunos educadores argumentan que deben quedar en los salones de clase regulares, mientras que otros recomiendan que

deben ser puestas en la biblioteca o centros especiales de medios. Así que en la mayoría de las escuelas orientadas a computación han optado por crear laboratorios especiales para aplicaciones EAC. Sin embargo con el reciente incremento en el uso de EAC para materias diversas, las computadoras encuentran su ubicación final en los salones de clases comunes y en centros de cómputo departamental e inclusive en locaciones centralizadas de libre acceso.

El papel del maestro en la implementación de EAC, inicialmente ignorado por los autores de este tipo de producto, esta recibiendo un creciente énfasis. Seymour Papert, que es el autor del lenguaje LOGO, recordó a los participantes del Tercer Congreso Anual de Verano para Computo Educativo , que el maestro y no la computadora es el que debe permanecer como centro del proceso educativo. Los padres están también aprendiendo el valor de EAC y están comprando computadoras y programas de aprendizaje para sus niños en casa. Los programas de entrenamiento computarizado (*computerized training programs*) se están convirtiendo en algo común en las empresas también.

Mientras que la mayor parte de los "expertos" están de acuerdo en que el software educativo se ha mejorado en los últimos años, muchos otros señalan que podía ser, todavía, mejor. La demanda por buen software educativo se esta incrementando día a día, dado que el numero de usuarios se extiende y más profesores encuentran la manera de incorporar EAC en sus curriculums (planes de estudio).

En esta breve historia de lo que ha sido EAC se ha mostrado que las cosas pueden cambiar rápidamente. Lo que ayer parecía imposible puede estar a nuestro alcance

hoy, y ser obsoleto para mañana. Las lecciones aprendidas al paso de los años son las mas recordadas, especialmente, por los desarrolladores de EAC.

1.2 ENSEÑANZA ASISTIDA POR COMPUTADORA Y MULTIMEDIA

SURGE HIPERTEXTO E HIPERMEDIA.

El termino *Hipertexto* describe la representación electrónica de texto que aprovecha las capacidades de acceso aleatorio de las computadoras de evitar seguir estrictamente la secuencia del medio que ofrece el papel impreso (Marchionini, 1988). *Hipermedia* extiende la representación no lineal y de acceso a gráficas, sonido, animación y vídeo en movimiento. El concepto de hipertexto ha sido discutido desde 1940 pero no ha sido posible sino hasta mediados de la década pasada, cuando los avances en hardware e interfaces para software "amigables" (*user - friendly*) hicieron posible el concepto de hipertexto e hipermedia. El primer producto comercial de hipertexto /hipermedia, para ser distribuido a gran escala fue HyperCard de la corporación Apple para Macintosh. Otros productos para IBM y compatibles, así como para la línea de computadoras Apple II pronto le siguieron, dando lugar a docenas de sistemas de hipertexto /hipermedia.

Los seguidores de hipertexto e hipermedia pretenden que éstos sistemas modelen una memoria de asociación de ideas exclusivamente humana y por lo tanto puede servir como una herramienta cognoscitiva muy poderosa (Conklin 1987). Los hiperdocumentos, que son aplicaciones creadas con sistemas de hipertexto / hipermedia pueden contener una amplia gama de ligas implícitas o explícitas que le

permiten al usuario progresar a través de ellas basados en una asociación aleatoria de ideas. Los usuarios por lo tanto pueden seguir sus propios intereses en lugar de sujetarse solo a opciones permitidas por un programa determinado.

Hipertexto e hipermedia pueden ser utilizados en muchas aplicaciones incluyendo aquellas de consulta de base de datos así como por supuesto en el ambiente de la EAC. Sin embargo estos sistemas de hipertexto / hipermedia son relativamente nuevos para decir exactamente que impacto tendrán en el desarrollo de software educativo, aunque lo anterior no le resta interés para esta aplicación, inclusive ofrece la posibilidad de expansión a gran escala del potencial de EAC. La última aplicación que se le ha dado al hipertexto e hipermedia ha sido, sin duda, su uso en las páginas WEB, y es con éste ejemplo con el que se aprecia realmente su valor como herramienta de interfaz que permite una muy libre "navegación".

Los eventos más importantes en la historia del desarrollo del cómputo educativo de 1950 a 1990 se sumarizan en el siguiente cronograma múltiple. En éste se muestra la relación entre los avances de hardware y las actividades significativas en el campo de EAC. Como puede verse el mayor número de eventos se registra en la gráfica en los años más cercanos a esta década, lo cual ilustra perfectamente la razón acelerada de cambio que ha caracterizado el desarrollo de la EAC.

1950	1960	1970	1980	1990
DESARROLLO DE SOFTWARE :				
	Primeros "Mainframe" minicomputadoras	Introducción de (1975) Microcomputadoras Apple II (1975) Atari (1978) IBM PC (1981)	Macintosh (1984) Apple IIc (1988) IBM PS/2 (1987)	NeXT Apple IIGx
ESFUERZOS MAS IMPORTANTES EN COMPUTO EDUCATIVO:				
	Primeros esfuerzos en Instrucción Programada	Evaluación Sistemática de Software	Surgimiento y desarrollo de CMI -----	
	Primeros trabajos con IAC	Computer Curriculum Corporation (IAC Drill and Practice) -----		HyerCard (1987) (primer producto hipermedia)
	Proyecto PLATO U. de Illinois	1a Gen. IAC	2a Gen. IAC	3a GEN. IAC
		Surge la industria "casera" IAC Entran al mercado de software grandes publicadores		
PUBLICACIONES PRINCIPALES				
	"Science of Teaching/ Learning" (1954)	"Good Frames and Bad" (1962)	"Mindstorms" (1980) "Nation at Risk" (1982)	"Power On: New Tools for Learning" (1988)
OTROS DESARROLLOS IMPORTANTES:				
	"Computer Science" emerge como disciplina	Empeza la buena difusión de Introducción a la Comput. dentro de la Instrucción	La Ley de Software de familia permite el copyright	El 68% de los padres de familia están a favor del uso de computadoras en la educación.
1950	1960	1970	1980	1990

Fuente: Computer - Aided Instruction, Price V. Robert, 1991.

1.3 ENSEÑANZA ASISTIDA POR COMPUTADORA EN MÉXICO

BREVE RESEÑA DE LA COMPUTACIÓN EN MÉXICO.⁸

La primera computadora electrónica que funcionó en México, y de hecho en Latinoamérica, fue la que instaló la Universidad Nacional Autónoma de México en julio de 1958; se trataba de una IBM-650, que utilizaba bulbos electrónicos como componentes lógicos y tenía memoria de tambor magnético. Aunque sus dimensiones físicas eran considerables, su capacidad de cálculo y almacenamiento era bastante inferior a una actual, aun si podía ejecutar aproximadamente 10,000 operaciones por segundo. Por el lenguaje que manejaba el acceso era restringido a una pequeña elite de investigadores, pero con el tiempo se extendió a estudiantes principalmente de las carreras de Física y Matemáticas.

El Centro de Cálculo de la UNAM adquirió en 1960 un segundo equipo, la Bendix G-15, ampliando el círculo de usuarios que integraba estudiantes y profesores de Ingeniería y Química y también alumnos y docentes del Instituto Politécnico Nacional. Posteriormente, con la adquisición de equipo más potente de las dos instituciones, el Instituto Politécnico Nacional creó el Centro Nacional de Cálculo (CENAC) que tenía su mayoría de usuarios de la ESIME quienes junto con egresados de la UNAM, formaron la generación de dirigentes de los principales centros de cómputo.

⁸ La microcomputadora como auxiliar didáctico en el aula. ILCE, COEIBA.

El tercer centro de desarrollo tecnológico del país lo constituyó el Instituto Tecnológico de Monterrey que también adquirió sus propios equipos. Los institutos tecnológicos regionales, la Universidad Autónoma de Chapingo y las de Nuevo León y Veracruz se incluyeron también en el uso y enseñanza de la computación; éste se extendió después prácticamente a todos los planteles de educación superior con la llegada de las microcomputadoras a México.

La aceleración del proceso, la falta de visión para reponer e incrementar oportunamente los cuadros de instructores e investigadores y la incapacidad técnica para determinar la importancia de las microcomputadoras en la enseñanza, motivo que las instituciones de educación superior perdieran su liderazgo en la materia, pues los profesores emigraron a finales de los setentas a la industria y al gobierno y la capacitación quedó interrumpida. El Instituto Tecnológico de Monterrey se puso a la cabeza en la difusión de la computación, al instalar toda una red de microcomputadoras para brindar oportunidades de acceso a sus estudiantes logrando avances en la formación de especialistas en sistemas.

En 1985, el gobierno dio los primeros pasos para utilizar la computadora como un medio auxiliar en el proceso de enseñanza - aprendizaje; y para este efecto creó el Programa COEEBA- SEP.

La Secretaría de Educación Pública SEP y el Instituto Latinoamericano de la Comunicación Educativa ILCE consideraron respecto a la EAC en México y su aplicación al sistema educativo lo siguiente: Así como el papel, el lápiz y los libros

fueron en su momento una herramienta versátil en el ámbito educativo ahora lo es la computadora, instrumento que ofrece mas recursos y mayor capacidad para almacenar información que los anteriores. Por lo anterior, es indiscutible el relevante papel que desempeñaría la computadora en la educación.

Ahora bien el Estado mexicano consciente de la necesidad de modernizar su sistema educativo, al establecer el Programa para la Modernización Educativa adquirió el compromiso de efectuar una actualización docente que permita al profesor ofrecer a los alumnos los elementos necesarios para que se incorporen como individuos activos al proceso productivo de nuestro país. Por lo tanto, dada la gran responsabilidad adquirida, las autoridades, a través de la Secretaría de Educación Pública, realizo una serie de acciones que permiten a los docentes obtener los conocimientos necesarios para ubicarse dentro del concierto de la vida actual. Una de esas acciones fue el Programa COEBA-SEP (Computación Electrónica en la Educación Básica) que cumplía con lo que en un principio se planteó: introducir la computadora como auxiliar didáctico en las aulas.

Este programa ha permitido el desarrollo de un modelo educativo que tiene la ventaja no sólo de proporcionar información dosificada por medio de los Programas de Computación Educativos, según el nivel que corresponda, sino también de establecer la interacción y la comunicación entre los elementos del grupo o equipo que esté aprovechando el recurso, aspecto importante dentro de los objetivos educacionales que se refieren a la formación del individuo. Dado que cada vez la cantidad de conocimientos es mayor, también es mayor día con día, la necesidad de especializar adecuadamente a los profesores en las distintas áreas del

conocimiento, de tal forma que apoyen el trabajo interdisciplinario, para que los alumnos desde edades tempranas aprendan a valorar el trabajo en equipo, asimismo, debe lograrse una motivación en el docente a fin de inducirlo a prepararse día con día y romper así las barreras generacionales que impiden una comunicación cultural y técnica plena tanto entre docentes y alumnos, como entre padres e hijos actuales, para que el profesor que prepara a los alumnos, tenga todos los elementos que le permitan proyectarse al tipo de vida a la que sus alumnos deberán incorporarse como individuos activos en el proceso productivo del país.

PROGRAMA DE COMPUTACIÓN ELECTRÓNICA PARA LA EDUCACIÓN BÁSICA

PROGRAMA COEEBA.

A partir de 1986 el Programa de Computación Electrónica para la Educación Básica COEEBA asumió el compromiso de generar una serie de acciones tendientes a proporcionar al profesor de un grupo, herramientas que le permitan manejar a la computadora como medio, como auxiliar y como material didáctico. Yolanda Campos⁹ en 1990, esperaba que para 1992 casi el 90% de las escuelas secundarias y el 10% de las primarias oficiales del país contaran con al menos una computadora para utilizarla como apoyo didáctico.

COEEBA propuso que junto con los equipos, se contemple la dotación a las escuelas de software educativo para las diferentes áreas y materias de estudio.

⁹ integrante del equipo "Libro electrónico"

Hasta la 1994, se elaboraron más de 50 programas para secundaria, 6 para primaria, 3 para preescolar y 3 para educación de adultos, para el equipo Micro SEP; y 25 de secundaria y 30 para primaria para máquinas PC.

Yolanda Campos integrante de un equipo de desarrollo del proyecto de educación matemática para primaria y secundaria comentó en una ponencia “[...] Estamos conscientes de que se carece aún de evaluación, de que existe otro tipo de programas que pueden contribuir aún más al logro de los objetivos de una formación integral, sin embargo, en la medida de las condiciones y recursos de los que se dispone y considerando la realidad de la escuela, se presenta esta aportación, con el compromiso de continuar buscando la manera de optimizar los trabajos, de mejorar nuestra labor y de seguir en la búsqueda de apoyos para ofrecer a profesores y alumnos herramientas que les permitan la superación de lo cotidiano”¹⁰.

El Instituto Latinoamericano de Comunicación Educativa (ILCE), a partir de 1985, fue el encargado de coordinar el proyecto COEEBA-SEP, este programa tuvo como finalidad proporcionar a las nuevas generaciones los conocimientos básicos para utilizar la computadora en la ejecución y solución de las múltiples tareas, problemas y por supuesto, para que con el auxilio didáctico de esta herramienta tecnológica de esta herramienta se elevara el rendimiento escolar en todas las áreas del conocimiento.

¹⁰ La computación en el proceso enseñanza aprendizaje de la matemática en la educación básica en México.

DESARROLLO.

El programa inició su fase experimental en 1986, introduciendo la computadora como una herramienta de trabajo y un auxiliar didáctico en las áreas de Español, Ciencias Sociales, Ciencias Naturales y Matemáticas. Durante dicha fase se efectuó una selección y definición de procedimientos.

En 1989, se llevó a cabo la fase de generalización del programa, incrementándose así la elaboración de PCE (Programas de Computación Educativa) hasta abarcar el primer y segundo grado de secundaria y, con carácter de prueba la elaboración de PCE de educación primaria.

FUNDAMENTOS.

Entre los fundamentos de éste programa están, de acuerdo al Programa de Modernización Educativa de 1989¹:

- Marco jurídico - político. Adecuar los contenidos y métodos a la sociedad actual y futura de los educandos. Promover métodos de enseñanza - aprendizaje que propicien en el educando actitudes de indagación, experimentación y gestión. Elaborar "paquetes técnico - pedagógicos que sustenten y enriquezcan la practica educativa.

¹ILCE - SEP. Computación Electrónica en la Educación Básica. "La microcomputadora como auxiliar didáctico en el aula".

- Marco pedagógico - conceptual. Las ventajas que ofrece la computadora dan un cambio cualitativo importante en las metodologías didácticas y en consecuencia mejoría en los resultados alcanzados en la educación. Mayor posibilidad para integrar la teoría y la práctica. Regularización de alumnos que no se encuentran al nivel de aprendizaje del grupo, mediante los PCE.
- Infraestructura institucional. Los centros COEEBA - SEP realizan entre otras actividades el diseño y elaboración de PCE, principalmente con el objeto de abordar temas de interés estatal.

ESTRATEGIAS

Entre las estrategias estaban la evaluación permanente del programa por medio de la sistematización y opinión abierta de profesores y alumnos para actualizar los PCE. Producción, a nivel nacional, de más PCE de nivel secundaria y primaria.

RESULTADOS

Entre los logros hasta 1992 estaban el cambio de computadoras 8088 a maquinas 286, lo que represento obviamente un incremento en procesamiento y almacenamiento. Entre los proyectos que se tenían estaba el continuar la capacitación de los profesores en el uso de la computadora como auxiliar didáctico. Elaboración de los PCE de apoyo didáctico que estimularan y contribuyeran a la integración pedagógica de la educación básica. Actualización de los PCE de acuerdo a las opiniones y sugerencias de los profesores.

Para 1994, año en que con el cambio de sexenio el programa terminó, el equipo con el que se contaba eran máquinas con procesadores 286 y con memoria RAM de 4MB en promedio.

Lo mencionado anteriormente acerca del programa COEEBA- SEP es una breve síntesis del mismo. La razón por la que este programa se ha incluido es que a partir de 1994, el programa terminó dejando a las escuelas secundarias con talleres y salones de clases con computadoras pero sin desarrollo de nuevos Programas de Computación Educativos (PCE) y sin cumplir los proyectos acerca de la aplicación de la computadora como herramienta de apoyo pedagógico.

El tutorial CELLA tiene parte de su justificación en la falta de continuidad en el desarrollo de software educativo para escuelas secundarias, pues a partir del término del programa no se ha desarrollado nuevo software (de manera oficial, ya que algunos profesores por su cuenta han terminado o iniciado nuevos proyectos). Además el tutorial está desarrollado de manera que el equipo con que cuentan las escuelas pueda soportarlo y en caso de que actualizaran sus equipos pudieran interactuar con otros medios (audio, voz, etc.).

EL CENTRO DE PROCESAMIENTO ARTURO ROSENBLUETH (CPAR).

El CPAR sostiene que la idea común de llevar computadoras sin otros elementos como el entrenamiento de maestros y los programas de cómputo, ha sido rebasada en gran medida. Discute la manera en que las nuevas tecnologías de la información

y la comunicación pueden tener un impacto real en las diferentes actividades que se realizan en las instituciones educativas cómo promover el uso y manejo diferente de información, permitir situaciones didácticas más enriquecedoras, favorecer la participación activa y la actualización del educando, facilitar la actualización y superación profesional del maestro, apoyar la experimentación e investigación y, en suma, mejorar y modernizar la educación.

Para acercarse a resolver estos requerimientos se necesitan docentes con una visión general de la información educativa, que estén motivados a explorar con ella y que además cuenten con las habilidades mínimas que les permitan operar con soltura las herramientas de cómputo. Por otro lado, para que estén en posibilidad de poner en práctica dichos conocimientos es necesario que tengan acceso a espacios donde a partir de sus experiencias formulen propuestas fundamentadas susceptibles de generalizarse, y participen en la producción de apoyos didácticos como el software educativo.

El modelo adecuado para crear estos escenarios es el laboratorio. El CPAR tiene la experiencia de crear 82 Laboratorios de Informática Educativa en las Escuelas Normales del país. En particular el proceso de llegar a una solución integral (el laboratorio) y la participación de docentes y directivos en la construcción y desarrollo del proyecto académico que sustenta las actividades entre y, de laboratorios.

El que no se hayan incluido aquí otras instituciones no significa que en el país no haya otro productor de software educativo además de la SEP. De hecho,

actualmente la producción de software educativo se ha incrementado de manera muy considerable en tanto en instituciones educativas como en compañías que producen con la finalidad de venta de software. La UNAM a través del Centro de Investigaciones y Servicios Educativos (CISE) desarrolló el diplomado "Usos educativos de la Computadora" enfocado a la producción de software educativo a través de equipos multidisciplinarios. Otra dependencia que trabaja en éste tópico es la Dirección General de Servicios de Cómputo Académico que ofrece una línea de especialización "Elaboración de programas educativos computarizados" y que se desarrolla en el departamento de "Productos interactivos para la docencia". Se hace énfasis en estas dependencias pues la metodología adoptada para el diseño y desarrollo del tutorial CELLA se tomo del diplomado "Usos educativos de la Computadora" y de la línea de especialización "Elaboración de programas educativos computarizados".

CAPÍTULO II

TIPOS DE EAC

En los inicios solamente dos categorías de EAC son reconocidas, los programas de ejercicios y los tutoriales (Suppes & Morningstar, 1972). Ahora, existen otras categorías de EAC, en la literatura referente al tema. Todas las funciones de enseñanza en computadora actualmente reconocidas se analizarán en éste punto. Aunque muchos programas de EAC no se pueden clasificar con exactitud, es útil considerar cada modo por separado para ilustrar las funciones de enseñanza de la computadora.

La EAC puede ser dividida en cinco categorías: ejercicio y practica, leccionarios, tutoriales, simuladores y juegos.

Los programas de ejercitación son aquellos en los cuales el estudiante desarrolla repetitivamente alguna actividad con el fin de memorizar hechos. Los programas de ejercicio son la forma más común de EAC, y son los más fáciles de "programar". Practicar EAC implica el usar la computadora para aprender procesos, procedimientos y habilidades.

Los leccionarios (pasa páginas) presentan "displays" secuenciales en su mayoría en forma de texto, esto puede incluir gráficos y movimientos. La principal diferencia entre la página impresa y el "display" en computadora es que la última es dinámica.

Los tutoriales presentan información para el estudiante en pequeños segmentos acompañados de frecuentes preguntas y retroalimentaciones, tanto como un tutor humano podría hacerlo con un estudiante. Los programas EAC tutoriales tienen mucho en común con las primeras técnicas de instrucción programada convencional, pero con una mejor motivación y capacidad de guardar los registros. La diferencia entre los programas de ejercitación y práctica, los cuales son diseñados para reforzar la instrucción que ha sido dada fuera de la computadora, y los programas tutoriales es que asumen mayor responsabilidad en cuanto a la enseñanza. Los tutoriales son mucho más complejos de "programarse" que los de ejercicio y práctica, pero pueden ser aún más eficientes desde el punto de vista pedagógico.

Los simuladores presentan una representación de algún evento real o imaginario por medio del ambiente de la computadora. Al estudiante le deja experimentar en este ambiente para aprender las consecuencias de posibles acciones. Antes que los estudiantes directamente lleguen a sucesos específicos los simuladores les permiten explorar y descubrir conceptos implícitamente. Las simulaciones son probablemente el tipo más complejo de EAC para "programar".

Los juegos educativos pueden ser ampliamente clasificados como simuladores de vida o juegos académicos. Los juegos simuladores de vida presentan un escenario en el cual el estudiante juega o toma decisiones dentro de las reglas del juego. Política, ciencia, sociología y administración de empresas son áreas comunes de aplicación de los juegos de simulación de vida. Los juegos académicos combinan un formato de juego tal como lotería, dados o juegos de cartas con ejercitación, práctica o aplicación de alguna habilidad académica tal como matemáticas u ortografía. Los juegos educativos van desde los más simples a los más complejos.

2.1 PROGRAMAS DE PRACTICA Y EJERCICIO

El modo clásico de EAC es el de práctica y ejercicio (*drill and practice*) y aproximadamente el 50% de los programas de EAC actualmente en uso, están clasificados como programas de ejercicio, sin embargo, su elevada producción se debe no a la buena calidad de producción, sino que se debe a la facilidad de producción y uso. Estos programas realmente no enseñan, en un sentido estricto de dar una explicación o impartir un conocimiento, pero se basan en la premisa de que los conceptos ya fueron aprendidos en algún otro lado. Aquí el papel de la computadora es continuar con la enseñanza. Desde un punto de vista de teoría del aprendizaje, el propósito de los ejercicios y la práctica en general es transferir el conocimiento de la "memoria de corto alcance" a "una memoria de largo alcance" en el cerebro, y ayudar al usuario a manejar esta habilidad adquirida en el momento necesario (Price, 1989).

La mayoría de las personas no puede retener por mucho tiempo datos que memorizan para un uso inmediato. Así que para poder retener por un largo tiempo datos y conceptos importantes, es necesario que estos sean trasladados de la "memoria de corto alcance" a la "memoria de largo alcance". Para que esto suceda, se debe usar repetidamente la información o asociarla con algún suceso significativo. Es precisamente el objetivo de ejercicio y práctica de EAC: ayuda a aprender y recordar por repetición y ejemplos.

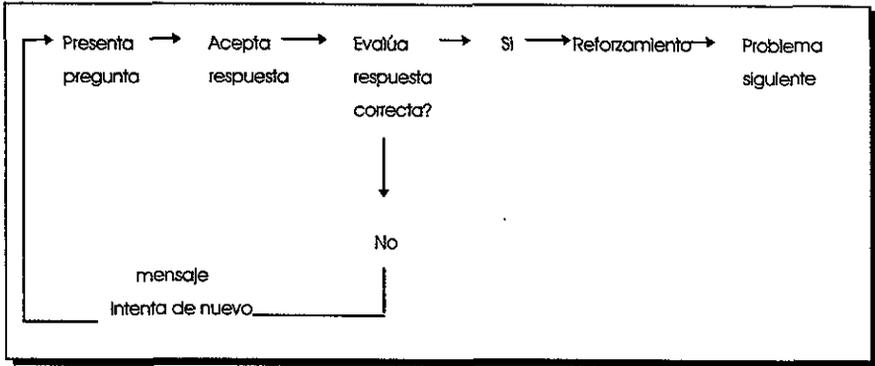
Es posible hacer una distinción entre programas para ejercicio y programas para práctica, aunque estos términos son comúnmente utilizados conjuntamente.

PROGRAMAS DE EJERCICIO.

Es el modo en que EAC desarrolla una actividad repetitiva para que el estudiante memorice ciertos hechos o datos. Algunos ejemplos de este tipo de programas son aquellos en los cuales los estudiantes son examinados con problemas de multiplicación, de conversión de temperaturas, de correcto deletreo de palabras, de identificación de estados y capitales y de memorización de definiciones. En los ejercicios de EAC, los problemas se generan por medio de programas de cómputo, mientras que en los métodos convencionales de enseñanza, los problemas son encontrados al final de los capítulos del libro de texto, o en hojas de trabajo separadas o en tarjetas de pregunta - respuesta rápida.

En cualquier método (convencional o por computadora) éstos ejercicios le ayudan al estudiante a recordar lo que ha sido aprendido. En el método por EAC, es la computadora la que compara la respuesta del estudiante con la respuesta correcta. Lo que ocurre después depende del programa de ejercicios en particular. Usualmente si la respuesta del estudiante es incorrecta, se repite la pregunta. En algunas ocasiones se dan algunas pistas o alguna explicación de tipo orientación. Después de dos o tres intentos fallidos la respuesta correcta generalmente se da a conocer, a veces con una breve explicación. Cuando la pregunta se contesta correctamente, generalmente se da una retroalimentación positiva, proporcionando frases como "buena respuesta", "Grandioso" o también "vas bien". Después de una respuesta correcta se presenta otra. En algunos programas se pueden seguir presentando preguntas hasta que se alcanza un número previamente determinado. En otros casos al estudiante se le da la opción de elegir cuanto quiere seguir

trabajando. La operación básica de un programa de ejercicios típico, se ilustra en el siguiente diagrama:



En el caso de programas para matemáticas, la computadora puede generar fácilmente números aleatorios para usarse en éstos problemas. Algunos programas generan los problemas de una lista previamente guardada dentro del programa o del mismo disco. Muchos programas son suficiente flexibles como para permitirle al maestro crear sus propios problemas. Otros son capaces de generar problemas de acuerdo a diferentes niveles de dificultad.

Algunos programas de ejercicio diagnostican y se adaptan a las respuestas según se desarrolle la sección, por ejemplo, si se da un gran número de respuestas incorrectas, el programa se puede autoajustar para presentar menos dificultad en las siguientes preguntas o problemas. De la misma manera si las respuestas correctas tienen un alto porcentaje, el programa puede cambiar a un nivel superior (de usuario). Algunas veces los programas de ejercicios incluyen gráficas y sonidos.

También es usual que timbres o tonos acompañen las respuestas correctas (en los programas para niños se pueden incluir también imágenes de colores), esto incrementa el interés por los programas, especialmente para los estudiantes jóvenes, y sirve como motivador, porque de esta manera el estudiante intenta obtener la respuesta correcta con tal de ver cual es la respuesta de la computadora.

Muchos programas de ejercicios llevan un registro del progreso de los estudiantes. Por ejemplo, el número de preguntas intentadas, número de respuestas correctas e incorrectas. El progreso se puede reportar en una tarjeta de aprovechamiento constantemente actualizada o mostrada periódicamente a través de la sesión. Algunos programas automáticamente graban el reporte de la sesión en disco, para que el maestro lo revise después. Otros programas proporcionan una impresión para que el estudiante guarde sus calificaciones cuando la sesión termine.

Los programas de ejercicio son los más fáciles de escribir, puesto que no asumen el total de la responsabilidad de enseñar, como se hace en otros modos de EAC. Además la simpleza relativa y la secuencia de avance de los ejercicios en EAC, requiere de menos sofisticación en el uso de la computadora. Los programas de ejercicios simples pueden ser escritos en tan solo unas líneas de algún lenguaje de programación básico como BASIC. Una vez que se tiene la impresión de que los programas de ejercicios son simples de escribir, se debe de mencionar que los programas de ejercicios realmente interesantes y útiles requieren de mayor habilidad.

Muchas de las críticas hacia la EAC han sido dirigidas precisamente a los programas de ejercicios, los cuales son frecuentemente acusados de ser tontos y carentes de imaginación. Esto puede decirse de cualquier técnica de ejercicios, no necesariamente de la asistida por computadora. Mientras es dudoso decir que en verdad los ejercicios en cualquier medio no son divertidos, es también verdad que es una técnica efectiva para enseñar materias exactas, la mayoría de lo que se enseña en foros educativos puede ser clasificado como material de ciencias exactas. Por ejemplo deletrear correctamente y la habilidad de multiplicar son por lo general, habilidades que resultan de la memorización. De cualquier manera que sean presentados los ejercicios (libro de trabajo, maestro o computadora) son medios efectivos de aprendizaje de este tipo de información y la EAC presenta varias ventajas básicas sobre las técnicas convencionales de ejercicios.

Los programas de ejercicios, en EAC, se basan en una teoría de aprendizaje fundamental y ampliamente aceptada. Pregunta, seguida por una respuesta del estudiante, seguida por una retroalimentación y un reforzamiento, es el proceso básico desde el punto de vista de aprendizaje de Skinner. La misma técnica ha sido empleada desde los comienzos de EAC. Y los resultados de investigación muestran que esta técnica es efectiva en términos de aprovechamiento del estudiante. A veces hasta la disfrutan. Supermath y Spelling Tutor son dos ejemplos de programas de ejercicios para Apple II.

PROGRAMAS DE PRÁCTICA.

Las prácticas en EAC involucran el uso de la computadora como un medio para proveer de procedimientos y habilidades en el proceso de aprendizaje. En los

laboratorios de ciencia, los estudiantes aprenden procedimientos para tratar ácidos y contar bacterias. En las clases de inglés los maestros tratan de enseñar procedimientos para construir frases correctas gramaticalmente. En el campo de las matemáticas los alumnos aprenden paso a paso procedimientos llamados algoritmos en la biblioteconomía se trata de aprender las secuencias y los pasos para localizar la información.

La práctica en el EAC está diseñada para proveer seguimiento práctico en procedimientos y habilidades que ya se han aprendido en cualquier otro lado. Así como los ejercicios de EAC, los programas de práctica proveen retroalimentación al estudiante de acuerdo a lo correcto de sus respuestas procurando algún ajuste del nivel de dificultad correspondiente. Sin embargo los programas de práctica hacen más que presentar simples problemas o preguntas. Muy a menudo estos requieren la negociación a través de un procedimiento complejo para poder llegar a una conclusión.

La ventaja de la práctica en cualquier situación de aprendizaje es obvia. La práctica hace la perfección. Y la práctica computarizada ayuda a que la perfección sea aprendida en forma conveniente. En muchas áreas de la vida real la práctica requiere de equipo costoso y del uso de ambientes de aprendizaje de limitado acceso. Para algunos tipos de aprendizaje los ambientes computarizados pueden ser tan buenos o mejores que otros medios de practica, y pueden ahorrarle tiempo al estudiante o permitir mayor práctica que la que se hubiera hecho de otra manera. La práctica en la construcción de algoritmos matemáticos, puede hacerse con papel y lápiz, pero los programas de práctica en EAC pueden juzgar inmediatamente si es correcta o no la formula dada por el estudiante, proporcionando además una

retroalimentación apropiada, o dando ciertas pistas y tal vez diagnosticando puntos débiles en su formación. Los programas de práctica son más complejos que los de ejercicio pero son menos complicados que los otros modos de EAC. La aplicación de programas de práctica esta también más limitada que cualquier otro modo de EAC. Solo un pequeño porcentaje de todos los programas de EAC cae en este modo. " *How Can I find it if Don't know what I'm looking for?*, *Reference Search* y *The food Processor* son ejemplos de programas de práctica, en los cuales no hay respuesta correcta, y fueron desarrollados para Apple II, y el último *The food Processor*, fue desarrollado para IBM PC también.

2.2 LECCIONARIOS (PASA PAGINAS)

Como el término implica en este modo de EAC se usa la pantalla de la computadora como si fuera una página en la cual la información se despliega tal como en la hoja de un libro.

Superficialmente esto podría verse como un uso muy pobre de la computadora, ya que el papel impreso es después de todo un medio muy conveniente que la computadora no estaría en posibilidades de remplazar, además, en comparación con las computadoras los libros son relativamente más baratos, fáciles de transportar y permiten un acceso aleatorio muy conveniente de su información. Las computadoras por otro lado son relativamente caras, tienen exhibidores visuales limitados, son más voluminosas que la mayoría de los libros y tienen una memoria limitada para el almacenamiento de texto. Sin embargo existen algunas cosas que hacen que la computadora enriquezca el proceso de pasa páginas.

La diferencia primaria entre la hoja impresa y la página de la computadora, es que el documento es estático y permanente, mientras que la página de la computadora es dinámica. Por ejemplo, se puede incorporar animación en las páginas de una computadora que despliega texto. Por ejemplo un diagrama animado que ilustre el proceso descrito por el texto, un cuaderno animado.

Cuando se incorporan en los sistemas de EAC equipos de videodisco o de videocinta se pueden incorporar al texto medios en color y movimiento, lo que da lugar a un cuaderno "en vivo".

Además por medio de la computadora se pueden cambiar porciones de la página para mandar imprimir, por ejemplo, en ilustraciones de procesos descritos como texto se pueden agregar otras pantallas por medio de transposición o ventanas. Las computadoras pueden ser también capaces de manejar muchos atributos de carácter tales como flasheo (blinking), texto en color, tipos y tamaños de letra. Mientras que todas esas características, excepto obviamente el flasheo, pueden ser incorporadas en trabajos impresos en EAC estas características pueden ser cambiadas por el programa según se necesite, mientras está cambiando de páginas.

Los programas pasa - páginas que incorporan estas características son raros, pero el legítimo uso de los pasa - páginas no debe de ser pasado por alto. Inclusive los pasa - páginas de tipo más simple que no hacen uso de las ventajas del potencial dinámico de la pantalla de cómputo, pueden todavía enseñar y ofrecer algunas ventajas. Considérese el hecho de una página impresa que no puede ser fácilmente cambiada, compita con una pantalla en donde las instrucciones de como desarrollar

un experimento, resolver un problema, etc.; puede ser fácilmente actualizada, expandida e inclusive corregida si estaba guardada en un medio magnético. Es por tanto claro que si tales instrucciones ya estaban impresas cualquiera de estos cambios requeriría de un elaborado y costoso proceso, incluyendo el tiempo invertido.

Las instrucciones que incorporan los pasa - páginas son comúnmente incorporadas en muchos tipos de software. Esto es valido si las instrucciones no son tan extensas. Una cantidad relativamente pequeña de información guardada en un disco puede ser más conveniente que esa misma información guardada en un manual. Muchos programas de aplicación no de EAC, tales como, hojas de cálculo, manejadores de bases de datos, procesadores de palabra, etc. incluyen sus propias instrucciones para el uso de si mismos (ayuda, demos, derechos).

Mientras que el uso de la computadora como un sustituto de el libro es probablemente una mala idea, los programas pasa - páginas tienen un lugar legítimo entre los modos de EAC. Un programa que ilustra el pasa páginas del modo EAC es el programa "Learn", el cual se proporciona con las computadoras mainframe VAX.

2.3 TUTORIALES

El diccionario Oxford define un tutor como "un maestro privado" o "alguien que esta a cargo de la educación en general de una persona". Los tutoriales difieren de los programas de ejercicios y de practica en EAC en el hecho de que asumen el papel de un profesor proporcionando nueva información mientras que los programas de práctica y ejercicios solamente proporcionan seguimiento de las enseñanzas y

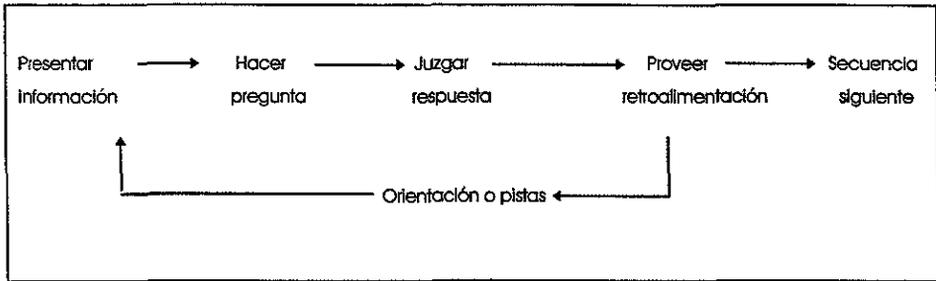
aprendizajes que se han dado aparte de la computadora. Muchos programas de ejercicios incluyen descuidadamente el término tutor o tutorial en su título. Dado que el modo tutorial asume un papel más a fondo en el proceso de enseñanza - aprendizaje, es el tipo más complejo de EAC y por lo tanto merece más atención en éste capítulo; además de que el producto de ésta tesis es un tutorial. Antes de continuar el estudio del tutorial de EAC, primero debe considerarse cómo enseña un tutor humano.

El tutor humano presenta en pequeños segmentos la información que el estudiante debe aprender y frecuentemente es convocado a responder cuestiones acerca de éste material. Después de que cada respuesta es evaluada se da una retroalimentación apropiada. Las respuestas incorrectas probablemente den lugar a la repetición de información relevante, quizás, con una aclaración adicional o ideas. Este proceso es llamado "retroalimentación correctiva". Cuando el estudiante responde correctamente, el tutor usualmente lo felicita y algunas veces refuerza la respuesta del estudiante ya sea repitiendo o reordenandola en una forma diferente. A esto se le llama "reforzamiento positivo". Si el estudiante tiene preguntas es libre de hacerlas y entonces el tutor juzga que tan bien esta aprendiendo el estudiante a través de sus preguntas y tal vez a través de su instinto. Para ayudar a asegurar la retención de la información el instructor puede de tiempo en tiempo, sumarizar lo que ha sido aprendido. Ejercitar o practicar sobre el material de estudio es incluido en intervalos regulares en el tutorial, si es que es apropiado al material que se está presentando. El tutor juzga cuando los niveles apropiados de aprendizaje han sido alcanzados y como esta respondiendo el estudiante. También es responsable de proporcionar descansos o el final de la sesión en forma apropiada.

Los tutoriales EAC imitan al tutor humano. La información es presentada por la computadora en pequeños segmentos, frecuentemente en forma de texto. También se utilizan apoyos visuales, de animación o de sonido en algunas ocasiones para aclarar nueva información. Los buenos programas tutoriales generan frecuentemente preguntas para el estudiante y evalúan las respuestas. Las respuestas incorrectas pueden tener como consecuencia la repetición de información relevante. Algunos programas tutoriales también proporcionan vías alternativas para intentar clarificar la información de aquellos estudiantes que tienen problema en aprender. Las respuestas correctas del estudiante son reforzadas por el programa de cómputo tal como si lo hiciera el tutor humano.

Los programas tutoriales pueden incluir sumarios o resúmenes y ejercicios o segmentos de práctica en intervalos regulares. Las pruebas de diagnóstico pueden ser proporcionadas en un programa tutorial para permitirle al estudiante saltar información que ya conoce. Los segmentos de prueba pueden ser también incluidos para evaluar que tan bien la información es retenida o que tan bien el estudiante puede combinar y aplicar la información que se presentó. En la mayoría de los casos de tutoriales de EAC, el estudiante controla su avance y si puede o no tomar descansos o salir en cualquier momento. Los tutoriales son mejores métodos de presentación de avance de información que los métodos convencionales de instrucción programada tratados anteriormente.

El procedimiento básico de un programa tutorial se describe en el siguiente diagrama:

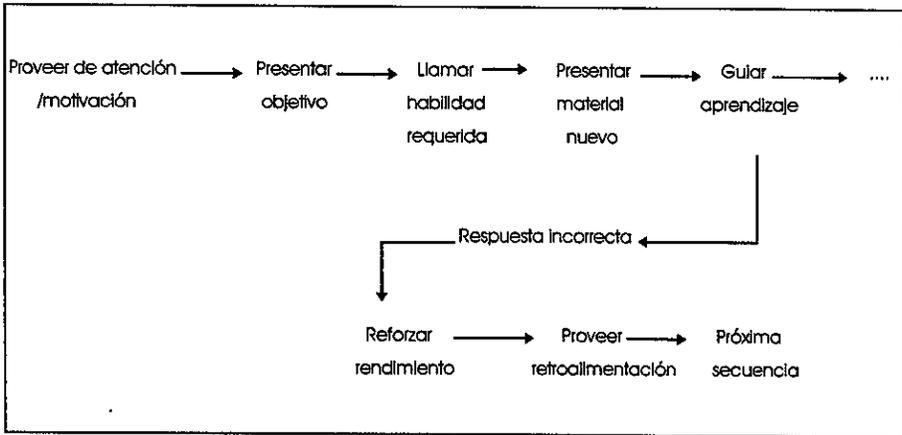


Este formato ha sido incorporado en algunos sistemas de autoría de manera que aquellos autores inexpertos puedan crear sus propias lecciones. Estos programas de autoría operan invitando al usuario a insertar el texto, las preguntas y la retroalimentación que serán dados. Un ejemplo es el programa "Genesis" elaborado por Bell y Howell.

El modelo anterior para instrucción tutorial sin embargo ignora algunas de las guías principales de la "buena instrucción" y por lo tanto los programas que lo siguen están perdiendo elementos importantes.

El procedimiento de este modelo no hace una previsión especial por motivar al estudiante o por encausar su atención y es por lo tanto lo más probable que sea percibido como aburrido o confuso. Tampoco se preocupa por comunicar los objetivos de cada lección al estudiante o por recordar los puntos relevantes a aprender. Esto significa que el estudiante puede no entender el propósito de la lección o no ser capaz de aplicar conocimientos adquiridos previamente a un nuevo material.

Además guías adicionales y pistas de orientación podrían ser útiles en el caso de respuestas incorrectas, referencias de texto y lecciones previas, así como la explicación de porque la respuesta fue incorrecta son algunos ejemplos. Wager y Rojas (1981) ofrecieron un modelo instruccional mejor para programas de EAC que incorpora estas consideraciones:



La planeación de las lecciones de EAC siempre depende de la naturaleza de los estudiantes y de los objetivos de la lección. La planeación de una pantalla específica para representar estos pasos es variable. Por ejemplo, el primer paso, "motivar la atención" puede implicar invitar al estudiante a un estado de alerta o referir al estudiante a algún material externo a la computadora. El segundo paso "presentación de objetivos" puede ser omitido si el diseñador estima que el estudiante potencial ya sabe cual es el objetivo.

PROGRAMAS LINEALES VS. PROGRAMAS RAMIFICADOS.

En general existen dos formas de programas tutoriales, el lineal y el ramificado. En la forma lineal al estudiante se le presentan secuencias consistentes de texto en las cuales a veces se combinan algunos apoyos visuales, preguntas y retroalimentación. Puede ser un número grande de secuencias las que compongan cada lección o capítulo, pero todas siguen el mismo formato, en los programas de este tipo todos los estudiantes siguen la misma trayectoria sin importar sus diferentes respuestas, es decir, estos programas únicamente están individualizados en el sentido de que el estudiante avanza a su propio ritmo y puede intentar varias veces la misma pregunta, y recibir alguna orientación o alguna pista entre respuestas.

Los programas ramificados tienen varias trayectorias educativas completamente diferentes. Por ejemplo el estudiante que equivoca varias preguntas puede ser mandado a una sección del programa con una secuencia de orientación que los otros estudiantes jamás ven. Algunos otros programas presentan opciones de tal forma que los estudiantes pueden elegir la porción de la lección o capítulo en la cual desean trabajar. Los programas ramificados también pueden hacer un examen previo al estudiante para ver que tanta de la información que será presentada en una secuencia dada ya era conocida por el, y probablemente omitir dicho material. Los programas ramificados son más difíciles de diseñar pero permiten mayor individualización, lo cual permite ajustar con más precisión las necesidades de cada estudiante.

COMUNICACIÓN.

El éxito de cualquier lección tutorial, ya sea humana o computarizada, depende de dos características:

- su disposición para comunicarse y,
- su habilidad para comunicarse.

Si cualquiera desea deliberadamente confundir el diálogo, el otro participante no tiene alternativa, pero si tan solo existe un espíritu de cooperación el acto de comunicación se vuelve más que un mero intento al concretarse. La probabilidad de una falla no intencional de comunicación en una sesión tutorial por computadora se eleva por las limitaciones de ambos. La computadora en sí y el autor del programa. Mientras la comunicación humana es enriquecida por inflexiones vocales, expresiones faciales y gesticulaciones, el tutor computarizado esta usualmente restringido a impresiones y gráficas. Esto obliga al tutor por computadora a usar estos recursos limitados con mucha mayor precisión y creatividad a manera de compensar la pérdida de otras importantes formas de comunicación.

Para un profesor (humano) cuestiones tales como dónde, cuando, cómo o a quien pueden resultar secundarias, en un sistema por computadora, no. Siempre que la computadora use una comunicación sin verbos se obtiene una mayor precisión para el uso de las herramientas restantes de comunicación requeridas.

La labor de programar una computadora para que actúe como un maestro se facilita si primero se trata de simular que el maestro actúe como una computadora. Puesto que este maestro imaginario puede actuar como una computadora, podemos asumir que tiene una memoria fantástica, sin embargo, será capaz de

recordar solamente aquellos puntos de información que han sido guardados en memoria por algún comando directo de nuestro programa. Para actuar como una computadora el maestro primero debe de convertirse en un pizarrón en blanco, y entonces cada pieza básica de información y su dirección de uso, deberá escribirse en el por medio de un detallado y cuidadoso plan de diseño de la lección. Si el profesor imaginario es confrontado por una pregunta sin la debida anticipación del estudiante, puede responder solamente dentro de los límites de la información que se le ha inculcado. Por lo tanto una forma de prepararlo para este tipo de preguntas es programar al maestro imaginario para que responda, con un conjunto de respuestas "para cualquier ocasión", tales como: "lo siento, pero no entendí su pregunta, podría repetirla por favor", si embargo se debe hacer lo posible por evitar este tipo de respuestas, porque si la computadora no puede continuar con un dialogo razonable, entonces el estudiante terminará frustrado.

En los buenos tutoriales de EAC es imposible progresar en la lección sin responder correctamente a las preguntas. Por eso es que es tan difícil completar un programa tutorial sin aprender muy bien el contenido. Para aquellos que experimentan dificultad para aprender, pueden sin embargo encontrar un programa de computadora que no se comporte como un tutor humano. Esto es, encarar al estudiante repetidas veces con las respuestas incorrectas, ya que los programas de cómputo pueden tomar una de dos opciones, ya sea repetir la información o permitirle al estudiante avanzar aun cuando no esta listo. Acerca de este dilema algunos programas simplemente detienen la ejecución y le dicen al estudiante que llame a un profesor o que pida asesoría.

Los buenos programas tutoriales son más difíciles de producir que los programas de práctica o de ejercicios. Tienen un mayor nivel de sofisticación puesto que deben asumir un papel más completo en el proceso de enseñanza. Un problema muy común en los programas tutoriales es el de la comunicación, lo que puede llevarlos a perder su objetivo, algunos otros problemas pudieran ser: fallas al examinar al estudiante, revisar apropiadamente sus respuestas, e incluso, fallas por el uso de las facilidades o capacidades de la computadora. Los siguientes son unos ejemplos de buenos programas tutoriales de EAC: *Budget Planner*, de Sun Belt Technologies para Apple II. *Problem - Solving Strategies: diagonals*, de MECC para Apple II. *Writing to read*, de IBM Education Systems para IBM PC.

2.4 SIMULADORES.

La simulación constituye uno de las más potentes aplicaciones de EAC. Los programas de simulación están diseñados para representar elementos esenciales de algunos eventos o fenómenos imaginarios o reales sin el problema de sus desaveniencias o peligros. El estudiante toma decisiones e inmediatamente puede ver las consecuencias por estas decisiones a través del uso de simulación en EAC. En ésta forma los programas de simulación le permiten al estudiante experimentar por sustitución que de otra forma no sería posible o al menos tan fácil. La capacidad de la computadora para realizar operaciones complejas a gran velocidad y proporcionar al estudiante una respuesta inmediata la hacen un medio ideal para el diseño y presentación de una simulación.

El amplio conjunto de ejemplos de este tipo de programas de simulación incluye simuladores de vuelo, imitadores de operaciones de la bolsa, réplicas

computacionales para procedimientos de laboratorio, réplicas de ambientes de negocios, simuladores de viaje y supervivencia, simuladores médicos para diagnóstico y prescripción, etc. Algunos simuladores están elaborados como un producto especializado mientras que otros tienen una apariencia física tal como la de un escenario de cómputo. Muchos tipos de juegos inclusive pudieran ser considerados simuladores, sin embargo para esta clasificación se consideran por separado éste tipo de juegos.

La mayoría de las veces cuando los profesores quieren que el estudiante explore una situación, pero encuentran imposible llevarlo al contacto directo con la misma ya sea por cuestiones de tiempo, logística, gastos y seguridad. Por ejemplo, considérese un problema obvio de enseñar la operación de un reactor nuclear. Un programa de simulación podría permitir al estudiante explorar en un simulador de reactor para experimentar y ver los resultados de sus acciones en la pantalla en lugar de verlas en la atmósfera del mundo real.

Un buen programa de simulación motiva al estudiante porque tal como un buen profesor lo inspira. Nada, excepto la situación real misma, le permite al estudiante explorar tan bien una situación como un simulador.

La mayoría de los programas con más éxito y populares dentro de los programas EAC son simuladores; algunos ejemplos de ellos son : "*Operation Frog*" de Scholastic Microcomputer Instructional Materials para Apple II, "*The Oregon Trail*" de Minnesota Educational Computing Corporation; y el más conocido "*Where in the world is Carmen San Diego?*" hecho en versiones para Apple o PC y distribuido por Broderbund Software.

2.5 JUEGOS EDUCATIVOS.

Todos aprendemos bastante de esas actividades llamadas juego, juegos de mesa, deportes, e incluso tareas de construcción que la gente hace "porque desea hacerlo". La asociación cultural de la palabra "play" ("juego") con placer es tan fuerte que es difícil, para la mayoría de la gente, asociar juego con actividades de aprendizaje. En realidad éstas pueden ser experiencias altamente motivacionales en una enorme cantidad de situaciones.

Existe una gran cantidad de juegos para computadora, cuyo principal propósito es el entretenimiento, y este es el motivo de que maestros y padres se quejen de grandes pérdidas de tiempo. Estos juegos pueden, de hecho, enseñar algunas habilidades y conceptos valiosos, incluyendo la coordinación ojo - mano. La importancia de seguir instrucciones y desarrollar estrategias para resolver problemas. Sin embargo, esto es diferente a que los autores de éstos juegos de entretenimiento intencionalmente busquen éstos resultados. Su propósito es, en cambio, dar únicamente entretenimiento, y si ocurre algún tipo de aprendizaje, ya sea importante o mínimo, esto es accidental. En contraste con los juegos de entretenimiento, los juegos educativos son planeados hacia resultados de aprendizaje definidos.

DESCENDIENDO AL JUEGO SERIO.

Aunque el juego esta muy asociado a la infancia, los atributos del juego también pueden ser encontrados en el aprendizaje adulto y aún las actividades de trabajo. Atributos como la observación, resolución de adivinanzas, y generar conclusiones

tentativas son básicas para jugar. Los “solucionadores de problemas”, adultos y niños, son igualmente confrontados con las situaciones de la vida real en las cuales la lógica detrás de la solución propuesta fue aprendida a través del juego. Habilidades y hechos pueden también ser aprendidos a través del juego. Mientras un estigma predeterminado es vinculado al concepto de juego como una instrucción seria, hay evidencia de que esto está cambiando. Los catálogos de software y las publicaciones muestran una variedad cada vez más amplia de material de juegos instruccionales en venta. Los maestros cada vez consideran más a los juegos instruccionales como un medio de instrucción y los libros de texto siempre incluyen, en Estados Unidos primero, y ahora en nuestro país también, actividades tipo juego en las listas de actividades y en los materiales de aprendizaje.

Un juego educacional es simplemente una actividad de toma de decisiones que usualmente incluye las siguientes características:

1. uno o más jugadores (tomadores de decisiones)
2. reglas del juego
3. uno o más objetivos que los jugadores tratará de alcanzar
4. condiciones dadas por oportunidad
5. espíritu de competencia
6. una estrategia de patrones de acción - decisión para ser tomadas por los jugadores
7. un sistema de retroalimentación para reconocer el estado del juego
8. un jugador o equipo ganador

Los juegos pueden tener uno o más jugadores y en los juegos computarizados la computadora puede también ser un jugador, si ésta ha sido programada para simular las acciones de un humano.

TIPOS DE JUEGOS EDUCATIVOS.

Los juegos educativos pueden ser clasificados dentro de dos amplias categorías:

1. Juegos de simulación de vida y
2. Juegos académicos.

Un tipo de juegos de simulación de vida es el juego de simulación de contexto, el cual presenta un escenario con una cercana o distante pertinencia a las actividades diarias o experiencias de trabajo. Estos juegos son usados en ambientes de aprendizaje tales como agricultura, ganadería, administración de pequeños negocios, entrenamiento militar y procedimientos de emergencia en plantas nucleares. Sin embargo estos juegos pondrán a los jugadores en muchas situaciones de toma de decisiones. Las reglas y objetivos del juego lo definen rigurosamente, el jugador no puede modificar las reglas o explorar alternativas interesantes que vayan más allá de las reglas.

En contraste con los juegos de simulación de contexto están los juegos de simulación de vida con final abierto. Ellos son usados en campos como la ciencia política, sociología, psicología social, y consultorías. En este tipo de juego es típica una libertad considerable. Los objetivos pueden ser modificados y las reglas que gobiernan el juego pueden ser cambiadas. Esta flexibilidad llama a la creatividad,

imaginación e innovación por parte del jugador(es). Además la libertad de los jugadores para manipular las restricciones básicas del juego los habilita para experimentar aspectos de los problemas no descritos en el escenario básico. Teniendo la creatividad por un lado y la dependencia del programa a la computadora por el otro; es claro que este tipo de juego de simulación no es fácil de "computarizar".

Los juegos académicos procuran combinar el formato de juego con el estudio de algunos tópicos menos atractivos. Ellos usan la emoción creada por jugar para alejar el tedio de algunas actividades repetitivas. En la mayoría de los juegos académicos el objetivo es proveer un desarrollo, ejercicio, aplicación o práctica de un concepto o habilidad que ha sido introducido por medio del juego. La enseñanza de estrategias de solución de problemas puede también ser un propósito establecido de los juegos académicos, pero esto es usualmente un riesgo difícil de superar. Hay numerosos ejemplos de juegos académicos en los cuales la materia de enseñanza es lo más importante.

Un estudiante que disfruta el "zapping" extraterrestres en la pantalla de video puede encontrar divertido un programa llamado "Word Attack". Aquí el estudiante puede usar un controlador de juego ("paddle") para "capturar" palabras que aparecen en la pantalla y colocarlas en su clasificación adecuada como sujetos, verbos u otras partes de la oración. El juego utiliza acción rápida y los marcadores son guardados con los mensajes de reforzamiento apropiados, desplegados por realizar un buen trabajo. En otro juego los estudiantes de primaria pueden practicar matemáticas con "Fay: Thah Math woman", quien camina, para, brinca sobre la recta numérica mientras responde las preguntas de los niños. El programa utiliza

sonido, gráficos y acción rápida; mientras los estudiantes compiten por los marcadores.

VENTAJAS DE LOS JUEGOS.

Jugar tiene muchas características que están asociadas con una buena instrucción. Primero, es imposible ser pasivo cuando se está comprometido en un juego (los espectadores no califican como jugadores). En el juego cada participante se identifica con el contexto, asume un rol y toma cierto grado de responsabilidad por las consecuencias. Los juegos estimulan la formación y evaluación de estrategias, el análisis de pautas principales y la exploración de los efectos de modificar estos principios. Los juegos siempre dejan algunos elementos de oportunidad o sorpresivos flashes de comprensión, características que son comunes a la vida en general y a la investigación y resolución de problemas en particular. Lo que es aprendido de los juegos está determinado por cómo y a qué grado están estructurados.

Diseñar programas de juegos EAC puede ser relativamente simple o extremadamente complejo. Los juegos académicos son generalmente más fáciles de escribir que los de simulación de vida, esto es porque los juegos académicos siempre son programas de ejercicio combinados con una estrategia de juego convencional. Los juegos usualmente utilizan gráficos y sonidos. Algunos juegos educativos de buen nivel son los siguientes: "Reader Rabbit an the fabulous Word Factory" esta programa combina gráficos y sonido y ayuda a niños de kinder y de primer grado a practicar asociaciones de palabras ."Mind Bind" es un paquete de tres juegos que

incluye un ejercicio de significado de palabras. Este software es para Apple y es de Princeton Software.

Algunos programas EAC no pueden ser clasificados exclusivamente como de ejercicio y práctica, leccionarios, tutoriales, simulación, o juegos educativos. Un programa puede ser primordialmente un tutorial pero también incluye componentes de ejercicio y tal vez use también un juego como premio. Otro puede combinar elementos de simulación con ejercitación o con un tutorial. La intención aquí no es hacer una clasificación rígida de todos los programas, sino prevenir a los productores que deben tener en cuenta todos estos modelos cuando estén planeando programas y usarlos independientemente o integrados de manera adecuada. Los tipos de EAC son clasificados por la manera en que tratan de instruir mediante la computadora.

Los programas EAC siempre contienen elementos de diferentes tipos de ella. Algunos pueden simplemente contener estrategias de enseñanza que no son incluidas en ninguna de las formas comunes (tutoriales, ejercicio y practica, simuladores, leccionarios, juegos educativos). Es inútil intentar clasificar los programas de EAC rígidamente por tipos. Ninguno de los tipos es mejor que los otros. Todos tienen su lugar en el medio del aprendizaje computarizado y cada uno tiene tanto aplicaciones adecuadas como ventajas y desventajas.

La forma en que la EAC ha sido clasificada está basada ampliamente en la manera de dar instrucción convencional. Debe considerarse muy seriamente que la EAC es un innovador medio que puede involucrar nuevos medios de instrucción que no

encajen exactamente dentro de los tipos establecidos. Los autores y productores deben ser motivados a experimentar y a no preocuparse por que sus programas encajen dentro de un tipo en particular.

Se debe mencionar también que la forma más reciente de educación por computadora, ahora con Internet, puede darse a través del WEB mediante programas educativos colocados en páginas; aunque sólo cambia la manera de "presentarlo" al usuario, ya que la esencia del software educativo se mantiene.

CAPÍTULO III

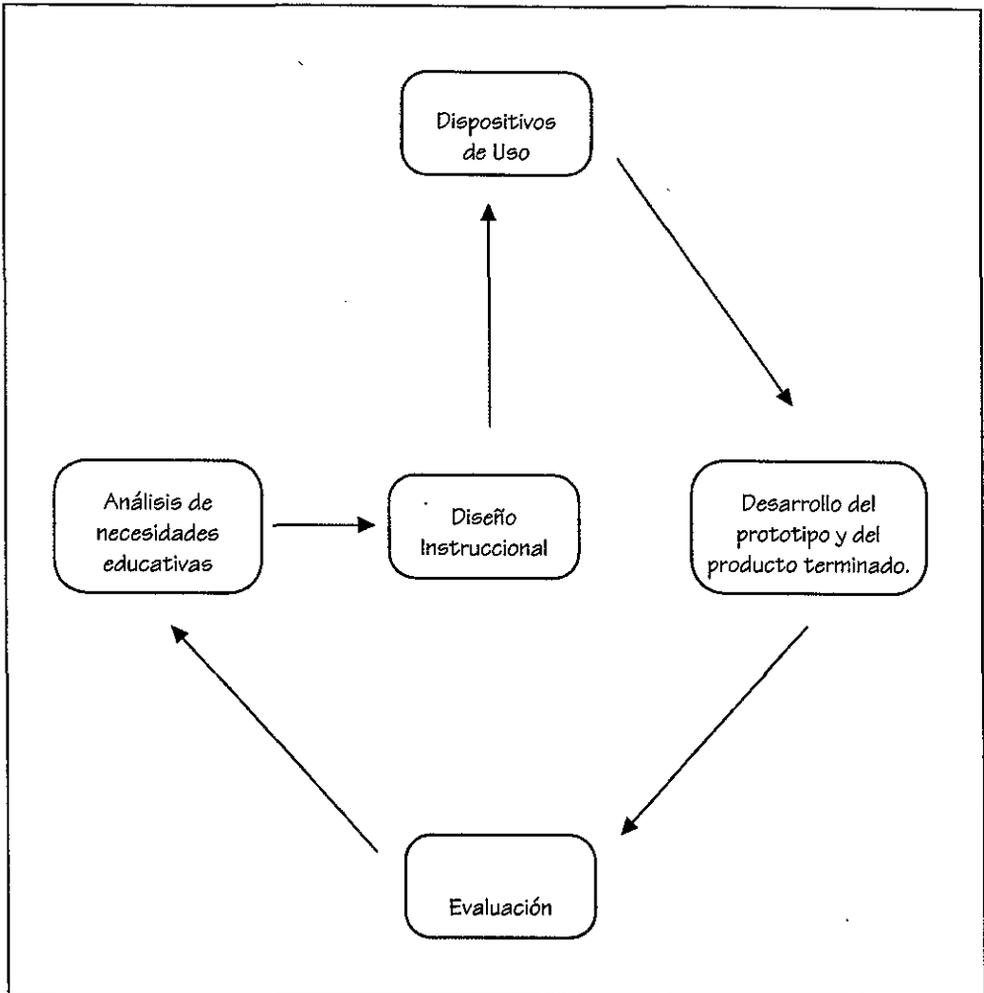
ANÁLISIS, DISEÑO Y PRODUCCIÓN

DEL TUTORIAL CELLA.

En éste capítulo se muestra el proceso de desarrollo del tutorial CELLA partiendo desde el análisis y terminando con evaluaciones en diferentes puntos. Es importante mencionar que la metodología empleada para el desarrollo de este proyecto es, en mayor parte, la adquirida en el diplomado "Usos Educativos de la Computadora" impartido en el Centro de Investigaciones y Servicios Educativos (CISE) y , en menor medida, la adquirida en la línea de especialización de Elaboración de Programas Educativos Computarizados de la Dirección General de Servicios de Compu Académico (DGSCA), ambas dependencias de la U.N.A.M.

La metodología para el desarrollo de Usos Educativos de la Computadora que se adoptó fue sistemática, esto es, basada en componentes interactuantes entre si, a partir de los cuales se rediseña el producto final. El esquema 3.1 representa la metodología adoptada.

Partiendo de un análisis de necesidades educativas del estudiante - usuario; y una vez investigadas y analizadas dichas necesidades, la solución factible se halla en el desarrollo de un programa de aprendizaje apoyado por computadora; posteriormente se continua con la planeación de las experiencias de aprendizaje que apoyaran el proyecto, o sea, la elaboración del diseño instruccional que constituye el eje del proyecto.



Esquema 3.1

Más tarde se estudia y decide que recursos de la computadora y/o equipo se requiere, tanto para el prototipo como para el producto terminado. El primer paso es el desarrollo del prototipo y del producto terminado, para concluir con una evaluación del programa para su corrección y/o extensión de uso.

El desarrollo de un producto de software educativo es un trabajo multidisciplinario. Es importante recordar que debe anteponerse la estrategia educativa y dejar en segundo plano los recursos de cómputo con que puede llevarse a cabo.

3.1 ANÁLISIS DE NECESIDADES EDUCATIVAS

Este punto aborda la primera etapa del modelo general de desarrollo presentado anteriormente: el análisis de necesidades educativas y el diseño instruccional. En ésta etapa se presentan las bases para el desarrollo de los usos educativos de la computadora.

Se presentan los elementos de juicio para que los desarrolladores decidan si elaborar un software educativo es la mejor opción. De lo contrario se proponen algunas alternativas de desarrollo como el uso o adaptación del software existente. En caso de considerar la opción de desarrollar software nuevo se trata el aspecto educativo que debe considerarse en la creación de software educativo. Por diseño instruccional deberá entenderse el plan, estructura y estrategia de instrucción dirigidos a la producción de experiencias de aprendizaje, con el propósito de lograr determinadas metas definidas con anterioridad.

La primer cuestión es decidir si se debe desarrollar software o no, y de acuerdo con M. Gándara¹² el desarrollo se justifica cuando:

¹²Usos Educativos de la Computadora, CISE, 1994.

- No existen soluciones disponibles
- Las que existen no cubren las necesidades detectadas
- Se puede competir contra las soluciones existentes y mejorarlas (aunque sea solamente en precio y accesibilidad).

Esto es así, dado que si ya hubiera soluciones satisfactorias, el desarrollo nuevo saldría sobrando, o no podría competir en condiciones favorables contra programas existentes; en este caso podrían usarse, adaptarse o modificarse programas ya existentes, o incluso hacer localizaciones¹³.

En el caso del desarrollo educativo, parece razonable que se cumplan criterios adicionales, en particular, que un nuevo desarrollo proceda cuando:

- Se ha mostrado que no hay soluciones disponibles y eficaces en otros medios, o sea, que la solución de cómputo es realmente necesaria y no redundante y superflua.
- La computadora realmente es el medio instruccional más adecuado para los objetivos de enseñanza - aprendizaje (es decir, logra lo que otros medios no podrían, aunque se cuente ya con materiales existentes).

¹³Localización en este contexto debe entenderse como el proceso de no solamente traducir un programa de su idioma original a otro, sino de hacer las conversiones y ajustes necesarios para que el programa tenga sentido en el contexto cultural nuevo en el que será usado.

Es de vital importancia que el desarrollo finalice en el uso, y no solamente en el almacén junto al proyector de cuerpos opacos o de cine de 16 mm. Esto requiere prever y asegurar que las condiciones y dispositivos para el uso estén presentes o puedan estarlo para cuando el desarrollo se concluya.

No todos los problemas educativos tienen solución óptima vía la computadora, (aún reconociendo en enorme potencial educativo del cómputo) primero debe definirse la necesidad o el problema, y solamente en relación a este, cual es el medio adecuado de solución, si la solución es la computadora, entonces hay que considerar si se justifica un nuevo desarrollo.

De ser necesario el desarrollo, y de acuerdo con la metodología que se sigue, se utilizará una estrategia de desarrollo estructurada o sistemática, es decir, la estructura global adoptará la forma de una organización jerárquica.

Ya que se definió una estrategia estructurada, empieza el proceso de desarrollo. Las etapas de este proceso son básicamente cuatro: el diseño, la instrumentación, la prueba y depuración final, y la entrega o implantación. Cada etapa tiene etapas o fases que en ocasiones no tienen una secuencia estricta o pueden ocurrir paralelamente. En el punto de Desarrollo del Prototipo y Producto Terminado se detallaran estas etapas.

Como es notable hasta éste punto el desarrollo de un producto de este tipo tiene que ser un trabajo interdisciplinario. Esta metodología sugiere trabajar en equipos formados por especialistas en el tema del contenido (en el área de conocimiento sobre la que trata el desarrollo), en pedagogía (que garantice que el programa

realmente apoye el aprendizaje), en computación (responsable del código), en diseño gráfico (que supervise el diseño de las pantallas y digitalizaciones), y un organizador (que administre el proyecto) ; que tengan la capacidad de trabajar en un equipo interdisciplinario.

La composición de este equipo puede variar según el proyecto y también según la etapa del proyecto. En el CISE se opto por una estrategia muy parecida a esta "estrategia ideal"¹⁴ reuniendo grupos interdisciplinarios en cada diplomado.

3.2 DISEÑO INSTRUCCIONAL

El desarrollo de usos educativos de la computadora generalmente se realiza al margen de un diseño instruccional adecuado, esto es, sin una planeación de las actividades de enseñanza y la sistematización de experiencias de aprendizaje que tendrán lugar. El objetivo de este punto es presentar las etapas que comprende un diseño instruccional.

El modelo de instrucción empleado está basado en el modelo de Dick y Carey (1978) con algunas modificaciones debidas a la experiencia y a la orientación explícita hacia el empleo de la computadora como apoyo para el aprendizaje. Este modelo se basa en facilitar el transito de la información de la memoria de corto plazo a la de largo plazo, desde la construcción del conocimiento hasta la recuperación y transferencia de lo aprendido.

¹⁴ Estrategia sistemática y con apoyo de un equipo interdisciplinario.

Generalmente la persona que elabora el diseño instruccional es el profesor que es el que está en contacto directo con los alumnos y conoce la problemática de la enseñanza. Se supone que esta persona tendrá pericia en el área de conocimiento y en la práctica docente.

Con esto definido, se presentan los pasos que comprende la elaboración de un diseño instruccional apoyado en el uso educativo de la computadora con algún contenido específico:

1. Análisis de necesidades educativas
2. Meta instruccional
3. Sistema de producción
4. Análisis instruccional
5. Objetivos de aprendizaje
6. Estrategia instruccional
7. Medios instruccionales
8. Evaluación del aprendizaje

1. ANÁLISIS DE NECESIDADES EDUCATIVAS.

A partir de este análisis se identifican las deficiencias del sistema educativo existente, y sus posibles causas y soluciones (Galvis, 1992). Entre las posibles soluciones el uso de la computadora resulta un excelente auxiliar del aprendizaje, por lo que debe analizarse cuidadosamente la conveniencia de tomar esta solución.

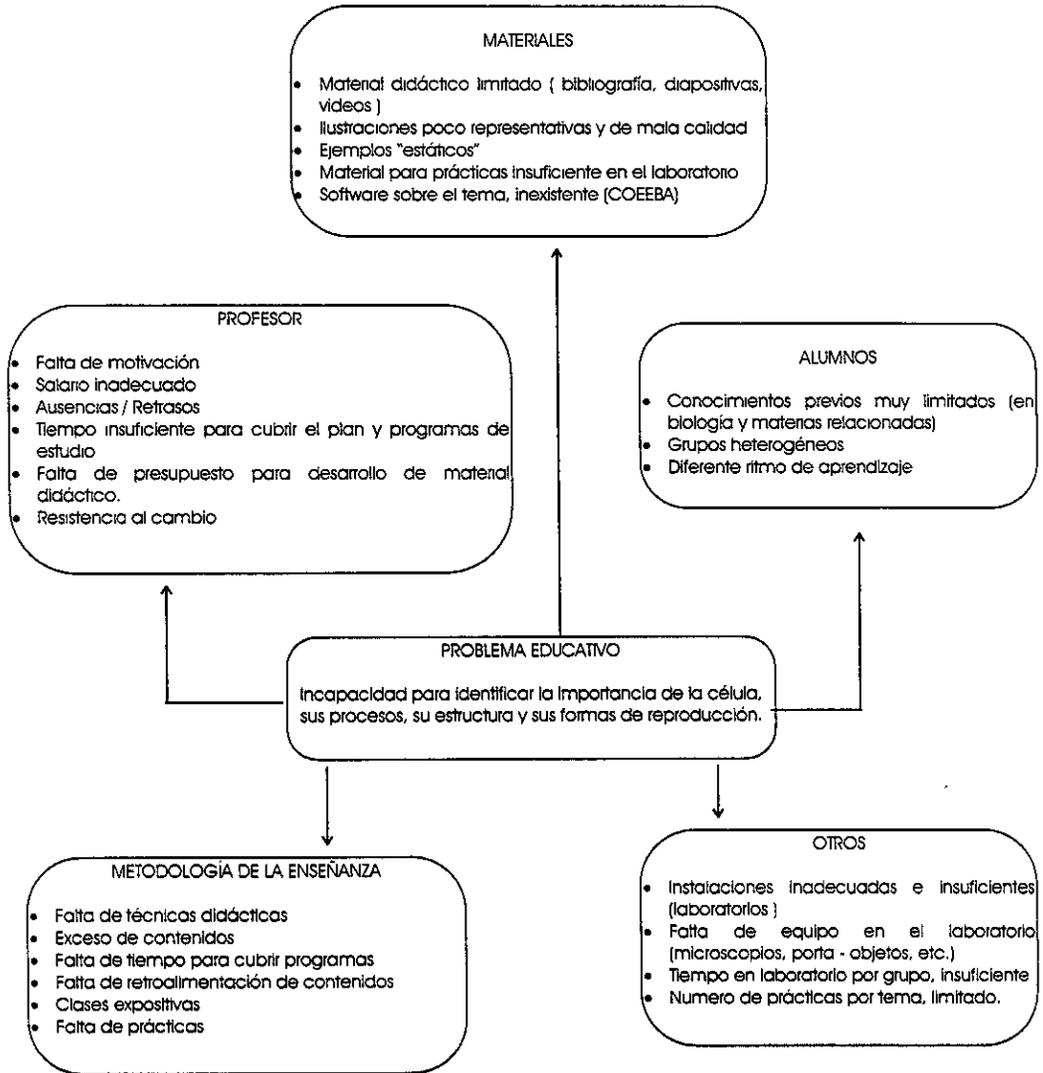
Analizar una necesidad educativa conlleva la identificación de algún problema de aprendizaje; para identificar un problema de aprendizaje debe preguntarse ¿Qué es lo que más trabajo les cuesta aprender a los alumnos?. Detectar un problema de aprendizaje es una tarea importante cuando se hace uso de la computadora como apoyo del aprendizaje. Es necesario centrarse siempre en el alumno, pues pensar en los problemas del alumno traerá resultados positivos en el aprendizaje del mismo.

Una vez identificado algún problema de aprendizaje, el siguiente paso es el análisis de la viabilidad del uso de la computadora. Esto se logrará descomponiendo las posibles causas del problema detectado. Para este fin se utiliza un esquema que contiene indicadores para cada elemento de análisis. El esquema 3.2 incluye los indicadores del proyecto CELLA.

Con la elaboración de éste análisis se reafirma la relevancia del proyecto; confirmar o refutar que la solución puede ser el uso de la computadora y el reconocimiento de un problema de enseñanza - aprendizaje multifactorial.

Como se observa en el esquema 3.2 el problema parte de la incapacidad de los alumnos para identificar de manera "real" la importancia de la célula y conocer su "forma de vida".

ANÁLISIS DEL PROBLEMA



Esquema 3.2

Los materiales didácticos con que cuenta la institución se reducen a bibliografía, en su mayoría obsoleta, laminas hechas por alumnos que van de regular a mala calidad, y en los casos en que el tema lo permite, apoyo de observaciones en microscopio dentro del laboratorio. Generalmente el tema se ha manejado a base de clases teóricas en el salón de clases con apoyo del pizarrón y en algunos casos de bibliografía con ilustraciones que en su mayoría no son representativas para los alumnos. Debe recordarse que las ilustraciones no ofrecen movimiento y que esta necesidad podría cubrirse con el uso del video, si lo hubiera. Sobre el uso de software educativo referente al tema, debe mencionarse que el programa COEEBA no desarrolló ningún PCE para éste tema.

Los alumnos poseen pocos y heterogéneos conocimientos previos (para el nivel de educación básica secundaria un pequeño tema en la primera unidad del primer año). Debe mencionarse que independientemente del nivel de conocimiento, cada alumno tiene su propio ritmo de aprendizaje.

Los profesores tiene grupos muy grandes lo que restringe el tiempo que pueden dedicar individualmente a cada alumno. El tiempo asignado en el plan de estudios es insuficiente para cubrir ampliamente todos los temas. Las instituciones no cuentan con presupuesto para desarrollo de material didáctico. Además de su situación socio - económica que crea apatía y resistencia al cambio.

En cuanto a la metodología de la enseñanza las clases se llevan a cabo con el sistema tradicional (clases expositivas) que implica exceso de contenidos, falta de tiempo, de retroalimentación y de prácticas. El tema por su naturaleza no permite, la mayoría de las veces, una observación "real" ni directa, esto obliga a los

profesores a explicar conceptos sólo con la ayuda del pizarrón, medio muy limitante y que exige mucha imaginación de parte de los alumnos.

Dentro de otros factores están: el tiempo en el laboratorio limitado, generalmente una hora a la semana, lo que restringe el número de prácticas por tema. Instalaciones inadecuadas (por la falta de bancos, cortinas, etc.) e insuficientes. La falta de equipo en el laboratorio también es muy importante (microscopios, porta-objetos, etc.).

2. META INSTRUCCIONAL

La meta instruccional es la elaboración del enunciado que expresa lo que el alumno estará en capacidad de hacer cuando termine el periodo de instrucción (Dick y Carey, 1978). Es importante, notar si la habilidad descrita es sólo una o más y si son independientes o están relacionadas.

Las metas instruccionales de este proyecto son:

- El alumno identificará e integrará la célula en niveles de organización superiores.
- El alumno valorará la importancia de la célula como un sistema vivo
- El alumno describirá la estructura básica de una célula estándar
- El alumno observará el comportamiento, estructura y reproducción de la célula, por medio de simulaciones de la computadora.

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

3. SISTEMA DE PRODUCCIÓN

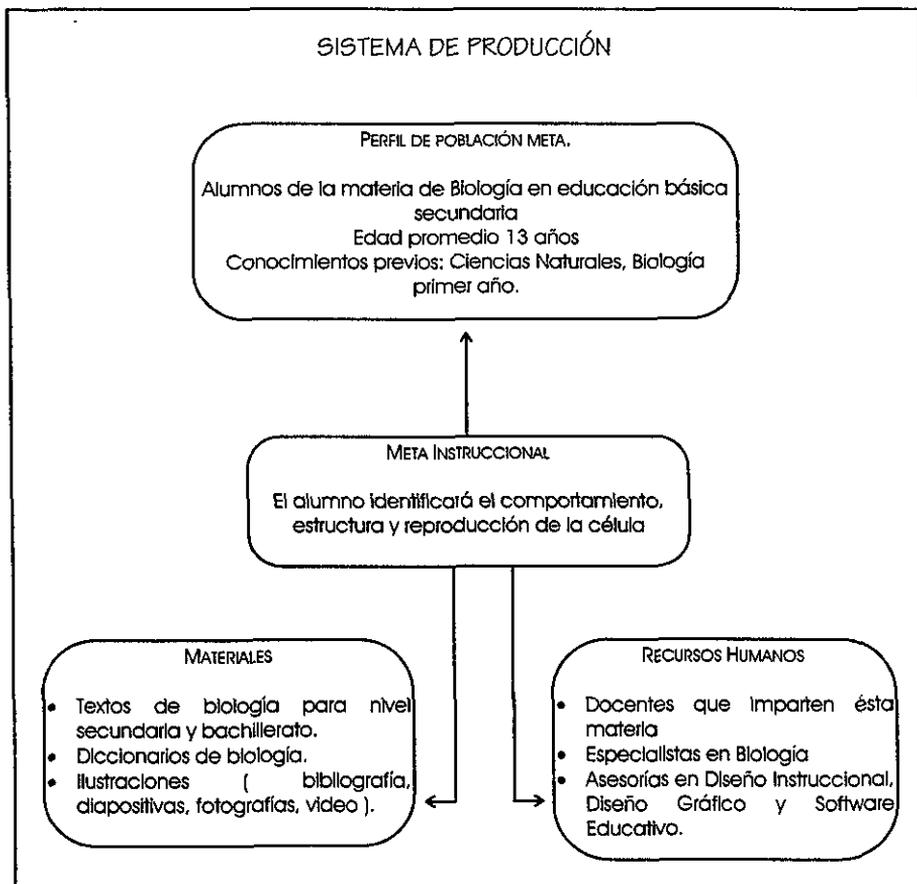
Una vez determinado que es lo que se propone que el alumno será capaz de lograr, el siguiente paso es determinar los recursos humanos que se necesitarán para el logro de la meta, los materiales que se requerirán, así como con los que ya se cuentan y la descripción de la población meta, es decir, la población hacia la que va dirigida la instrucción. Los indicadores del proyecto CELLA se muestran en el esquema 3.3.

La identificación de los materiales es necesaria para considerar aquellos relacionados al contenido de instrucción que pueden servir para preparar lo planeado, o bien para buscar aquello que se necesitará para cubrir la meta propuesta (así como los recursos humanos) . Esto es importante sobre todo en proyectos de corte interdisciplinario.

La descripción de la población meta debe darse a dos niveles, por un lado mencionar aquellas características generales de interés o relacionadas a la meta instruccional y por otro a la descripción de los conocimientos previos.

Este punto es el más relevante ya que para diseñar cualquier tipo de instrucción se debe partir del nivel de conocimientos y habilidades que el sujeto posee al iniciar, a partir de ahí, propiamente la instrucción o decidir que conocimientos previos reforzarán al inicio de ésta.

El producto de ésta etapa es un análisis de viabilidad del proyecto educativo que se pretende realizar, que tan factible es, no sólo académicamente sino en el personal necesario para llevarlo a cabo¹⁵.



Esquema 3.3

¹⁵En recursos humanos, debe recordarse que el proyecto, de acuerdo con la metodología adquirida en el diplomado "Usos Educativos de la Computadora", es de carácter interdisciplinario.

4. ANÁLISIS INSTRUCCIONAL.

Hasta este punto se ha puesto en claro un problema específico de aprendizaje y su resolución a través de la computadora (análisis de necesidades). Con base en esto se tiene el planteamiento general de lo que se desea que el alumno sea capaz de lograr al término de la instrucción, es decir, la forma de solucionar el problema identificado (meta instruccional). También se han planteado algunos de los elementos que se necesitarán para llevar a buen término la meta propuesta, se han identificado los materiales existentes que apoyarán el desarrollo, que materiales y recursos humanos se requieren y que conocimientos y características generales tiene la población a la que va dirigida la instrucción (sistema de producción).

Así pues se tiene ya el perfil de entrada (conocimientos previos) y el perfil de salida (meta instruccional), y entre estas dos condiciones existe la tarea de descomponer la meta instruccional en todas las tareas de aprendizaje subyacentes, detallándolas hasta el punto que se identifiquen las habilidades que es necesario construir, como complemento a las que ya trae el alumno. Esta tarea se conoce como el análisis instruccional y se refiere a al procedimiento para conocer las habilidades o conocimientos subordinados necesarios para lograr la meta de instrucción propuesta.

Este es el paso más importante dentro de la planeación instruccional, pues aquí se desprenden: la enunciación de los objetivos de aprendizaje, de las estrategias instruccionales, de los medios instruccionales y de la evaluación que llevará a la verificación de la meta instruccional. Llegando a este punto sobresale nuevamente la pericia del profesor que planea el diseño instruccional, éste es quien conoce la

estructura de todo el contenido y quien ha tenido la experiencia de impartirlo. Para el proyecto CELLA, éste elemento tan importante fue la profesora de educación secundaria Martha Luján, ella imparte la materia de biología en segundo y tercer grado.

La importancia del análisis dentro de la elaboración instruccional consiste, primero, en que permite conocer con certeza los conocimientos previos necesarios e innecesarios para lograr la meta instruccional; segundo, permite realizar una selección, secuencia y organización del contenido en la medida en que se presenta un panorama de la secuencia lógica de los pasos que llevarán al logro de la meta de instrucción; tercero, la representación gráfica de la de la estructura general que tendrá el programa educativo. Esto es importante tenerlo desde éste momento, ya que permitirá empezar a decidir si el estudiante/usuario se moverá a través de una secuencia lineal, o si tendrá acceso a menús o ayudas.

Para realizar el análisis instruccional existen tres métodos¹⁶:

- Método Procedimental. Descompone la secuencia de instrucción paso a paso. Se emplea cuando la conducta a la cual debe llegar el estudiante consta básicamente de una serie de habilidades que deben de ser expresadas en secuencia para lograr la meta propuesta.
- Método Jerárquico. Este método también consiste en descomponer paso a paso la tarea. La diferencia es que no existe necesariamente una secuencia rígida y/o lineal para alcanzar la meta instruccional.

¹⁶ Dick y Carey (1978)

- *Método combinado.* En éste caso se conjuga tanto el método procedimental como el jerárquico su uso se recomienda cuando la meta instruccional hace referencia a habilidades cognitivas complejas, es decir, aquellas que requieren una secuencia, y las que además necesitan habilidades subordinadas.

En realidad, no es posible seleccionar inicialmente un método y desarrollar el análisis instruccional. El método a utilizar se obtiene finalizada la representación del análisis, por lo general el tipo de aprendizaje a descomponer dará la pauta para su esquematización.

Para el proyecto CELLA se tomó como base el plan de estudios elaborado por la Secretaría de Educación Pública.

PLAN DE ESTUDIOS

Para la SEP, en el nivel medio básico, el propósito general de la enseñanza de la biología es promover el conocimiento de los alumnos sobre el mundo viviente; sin embargo, los beneficios de una educación científica no deben limitarse a la adquisición de conocimientos. La ciencia es también una actividad social que incorpora valores y actitudes; su practica y el aprendizaje de sus métodos propicia la aplicación sistemática de actitudes como la diligencia, la imparcialidad, la imaginación, la curiosidad, la apertura hacia nuevas ideas, la capacidad de formular preguntas y, muy especialmente, debe inculcar en el alumno un cierto escepticismo sistemático que le permita balancear la aceptación indiscriminada de nuevas ideas.

La propuesta curricular plasmada en el Plan y programas de estudio 1993, como respuesta a las necesidades de la adolescencia y de la sociedad mexicana en general; además de estimular el interés por la actividad científica, promueven en el alumno actitudes de responsabilidad en el cuidado de su salud y del medio ambiente. Es ampliamente reconocida la importancia de mantener el equilibrio de los procesos ambientales y de contribuir a la conservación de la diversidad biológica. Por otra parte, los jóvenes están, a través de los medios de comunicación y de otros medios de enseñanza no formales, en permanente contacto con información sobre el estado del ambiente. En general, las experiencias cotidianas del alumno y de su percepción viviente deben ser punto de partida para el aprendizaje de la biología.

En esta propuesta la asignatura de biología presenta dos grandes niveles de aproximación. A diferencia de los programas anteriores, en primer año se estudian básicamente los procesos macrobiológicos, tales como evolución, ecología y genética. Con esta base, es posible abordar los conocimientos de biología del segundo grado que permiten comprender las particularidades de la organización de los seres vivos y su funcionamiento de manera general, analizando su fisiología y su anatomía.

“ ... En la unidad “La célula” se presentan elementos generales acerca de los procesos celulares. Principia con un análisis histórico sobre la teoría celular y los primeros trabajos acerca de la célula. En seguida se hace la revisión de los diversos tipos de células que componen a los seres vivos, con respecto a su estructura y a su función . Se analizan de manera general los procesos fisiológicos de la célula y se concluye con un análisis de la división celular.”¹⁷

¹⁷ SEP. Plan y programas de estudio 1993. Educación Básica Secundaria, p.57

SECRETARÍA DE EDUCACIÓN PÚBLICA

**EDUCACIÓN BÁSICA SECUNDARIA
PLAN Y PROGRAMAS DE ESTUDIO 1993**

Segundo grado

...

La célula

° Desarrollo histórico del concepto de célula

- Los trabajos de Robert Hooke
- La teoría celular de Schleiden y Schwann
- La célula: unidad anatómica, fisiológica y de

origen de los seres vivos

- Células procariontes y eucariontes
- Diferentes tipos de células en el cuerpo

humano

° El sistema membranal

- La membrana celular y sus funciones
- Alimentación celular : endocitosis, vesículas y

lisosomas, exocitosis

- La membrana nuclear y sus funciones
- El retículo endoplásmico, los ribosomas y la

síntesis de proteínas

- Aparato de Golgi y secreción

° El citoplasma

- Las mitocondrias y la respiración celular
- Los cloroplastos y la fotosíntesis

° El núcleo y la división celular

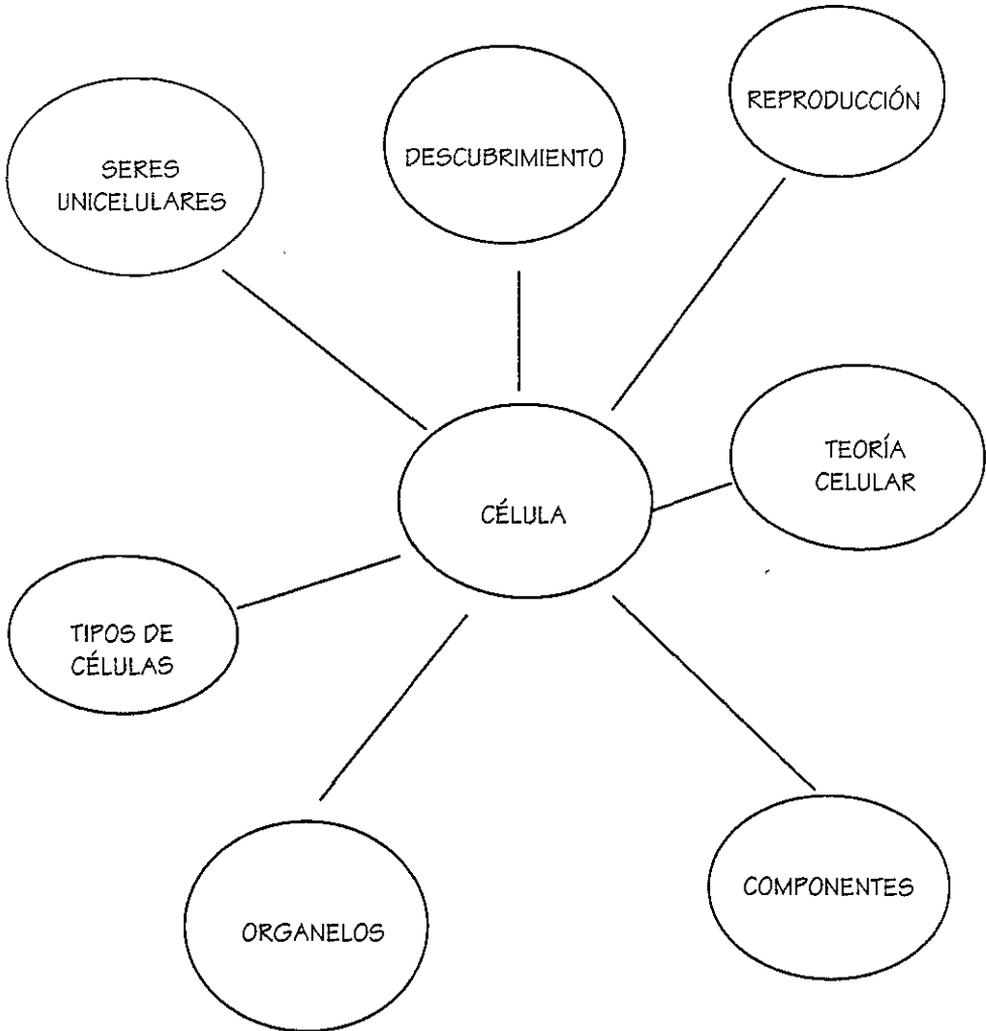
- Los cromosomas
- La mitosis
- La meiosis
- El ADN y la replicación
- El ARN y la transcripción

...

Fuente: Plan y programas de estudio 1993.

Educación Básica Secundaria. SEP

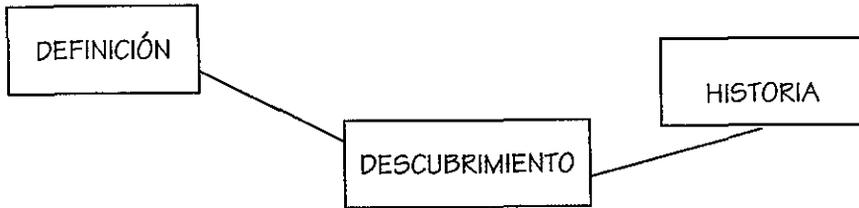
De acuerdo a éste plan de estudios se elabora un mapa conceptual que incluye los "conceptos" que el alumno debe dominar para lograr la meta instruccional definida con anterioridad.



Esquema 3.4

NIVEL 2

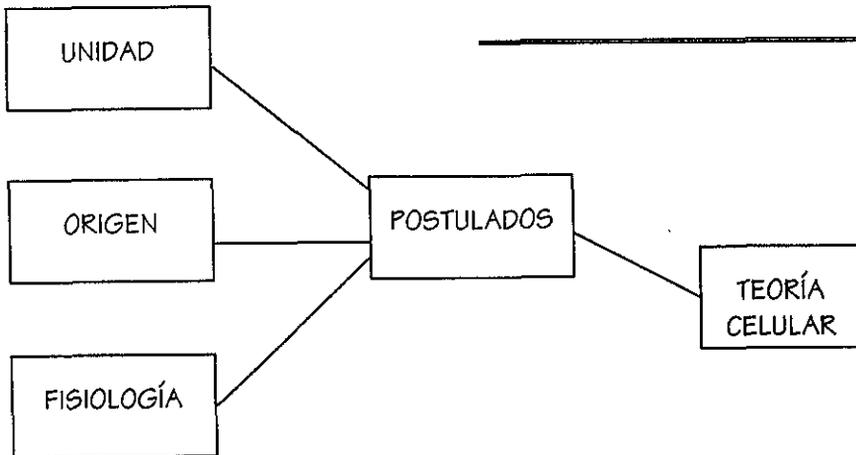
DESCUBRIMIENTO



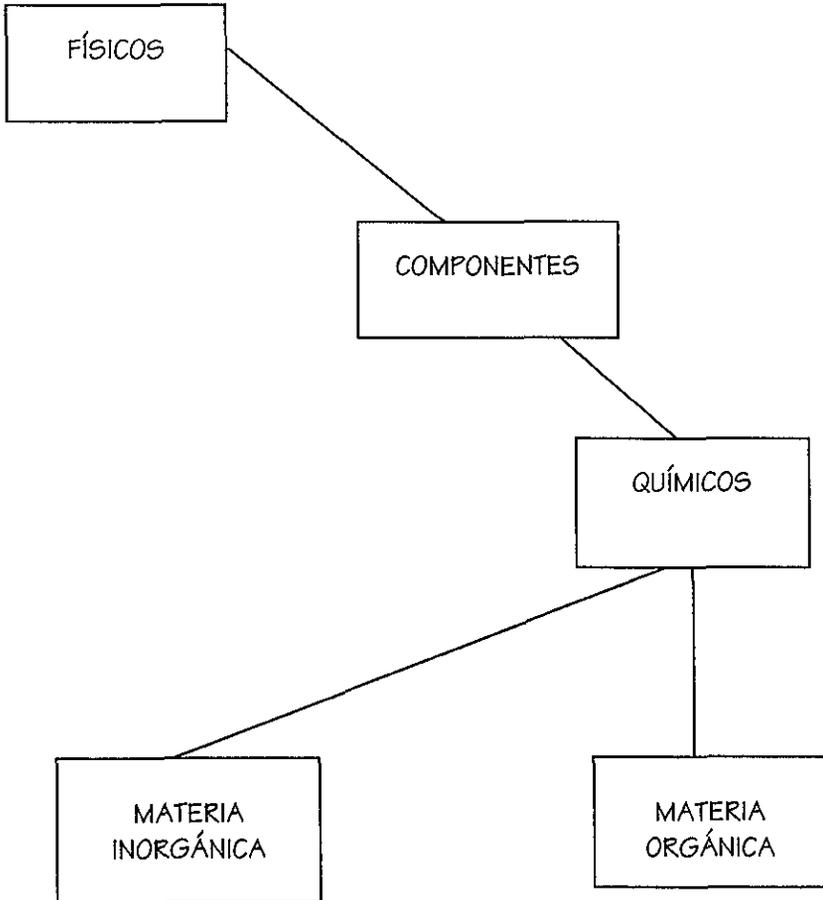
Esquema 3.5

NIVEL 2

TEORÍA CELULAR



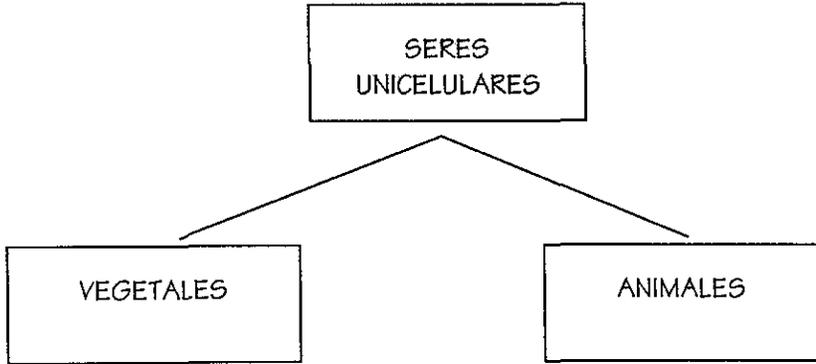
Esquema 3.6



Esquema 3.7

NIVEL 2

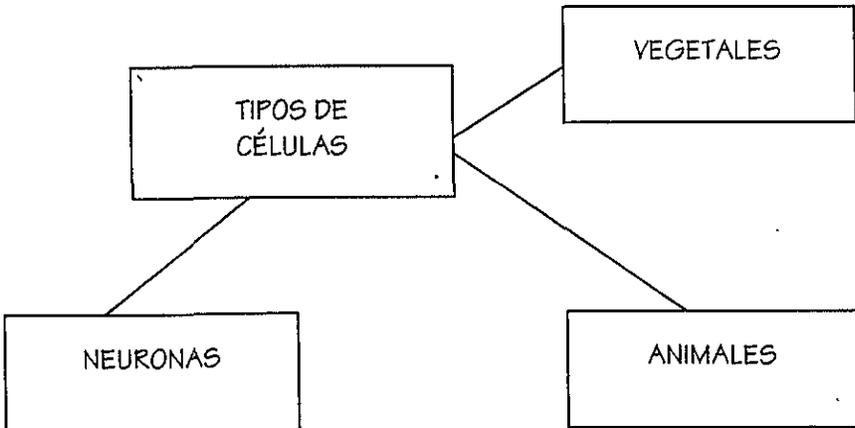
SERES UNICELULARES



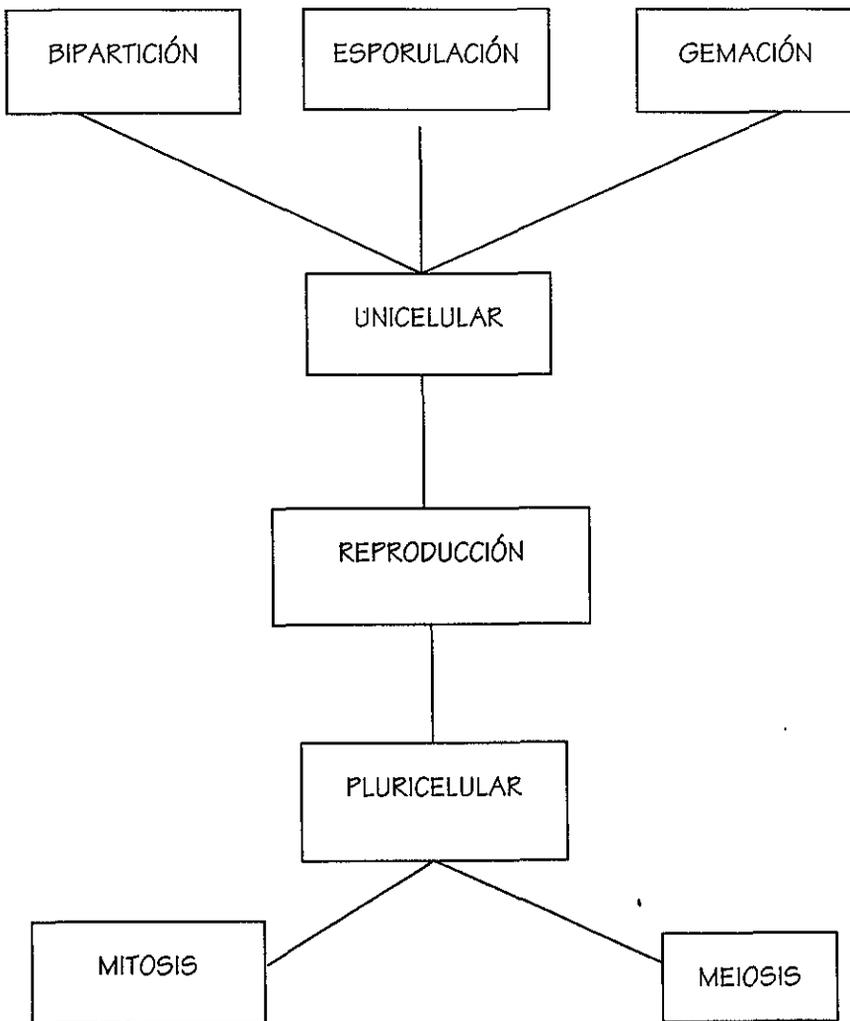
Esquema 3.8

NIVEL 2

TIPOS DE CÉLULAS



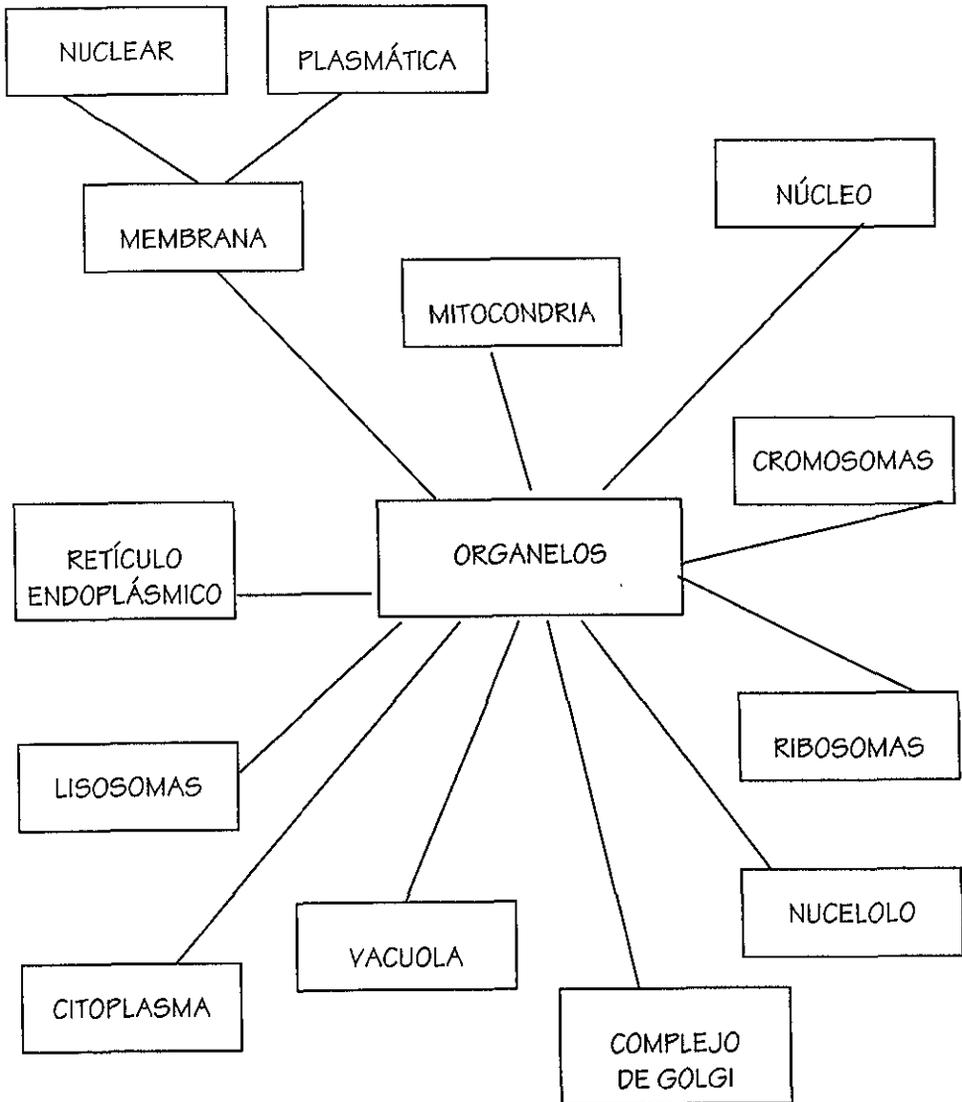
Esquema 3.9



Esquema 3.10

NIVEL 2

ORGANELOS



Esquema 3.11

5. OBJETIVOS DEL APRENDIZAJE

Una vez elaborado el análisis instruccional, el siguiente paso es redactar con base en dicho diagrama, los objetivos de aprendizaje. El objetivo del aprendizaje se refiere a un enunciado que explica lo que el alumno estará en capacidad de hacer cuando termine la instrucción, la diferencia con la meta instruccional es que en ésta el planteamiento es a nivel general sobre toda la instrucción, mientras que el otro se trata de la redacción de objetivos de aprendizaje específicos; de ahí la necesidad de que surjan a partir del análisis instruccional, ya que sin éste no se conocen con precisión el contenido ni la secuencia a seguir.

De acuerdo al análisis hecho en el punto anterior se redactarán los objetivos de acuerdo a los temas definidos:

- Dada la historia del descubrimiento de la célula y su definición, el alumno será capaz de deducir su importancia.
- El alumno podrá enunciar los postulados que conforman la Teoría celular.
- El alumno reconocerá los dos tipos de componentes en la célula (físicos y químicos), así como su función e importancia en los procesos celulares.
- Dada la definición de los organelos celulares, el alumno podrá identificar cada organelo y sus funciones en la célula.
- Dadas las características de células vegetales y animales, el alumno podrá identificar los organelos y funciones de cada tipo de célula.

- Dadas las características de las células vegetales y animales el alumno podrá identificar los seres unicelulares según su tipo.
- El alumno será capaz de relacionar el tipo de organismo (unicelular o pluricelular) con su forma de reproducción, así como explicar las etapas de cada forma de reproducción.

6. ESTRATEGIA INSTRUCCIONAL.

Una vez delimitadas la meta general, el contenido y la secuencia de lo que el alumno va a aprender y los objetivos del aprendizaje que enuncian lo que se alcanzará terminada la instrucción. Lo que sigue es el establecimiento de las estrategias instruccionales, esto es, de las decisiones sobre los componentes y procedimientos para lograr la instrucción adecuada al objetivo de aprendizaje propuesto.

Un elemento a considerar en el desarrollo de procedimientos instruccionales los componen los métodos de enseñanza, entendiendo a éstos como el procedimiento o el plan general de acción para lograr la meta instruccional propuesta.

Sobre los métodos de enseñanza existen varias tipologías. Aquí se tomará aquella que esté relacionada con los tipos de aprendizaje declarativo y procedimental: los métodos expositivo y por descubrimiento. El método de enseñanza por exposición es útil para el aprendizaje de conocimiento de tipo declarativo. La relación educativa básica es de profesor - alumno. Es el método denominado también como deductivo, donde se parte de conceptos para llegar a ejemplos.

El método por descubrimiento es útil para el aprendizaje de conocimiento de tipo procedimental. La relación educativa básica es realidad - alumno, también es conocido como método inductivo, donde a partir de ejemplos se puede llegar a conceptos. Es esta forma, en tanto el método de enseñanza se refiere a un plan general de acción para el logro de dicha meta. A continuación se mencionan algunas estrategias instruccionales útiles para el logro del conocimiento de tipo declarativo, así como el procedimental. La decisión sobre que estrategias emplear dependerá del tipo de método de enseñanza elegido, del tipo de conocimiento del que se trate, del contenido específico y del alcance del objetivo de aprendizaje propuesto.

El proyecto CELLA sigue un método de enseñanza expositivo para alcanzar conocimiento declarativo; esto se decidió conforme al plan de estudios de la SEP en el que está basado el contenido.

Estrategias para conocimiento declarativo.

- Presentar imágenes (ilustraciones, animaciones) siempre y cuando el material resulte familiar y las sugiera.
- Analogías cuando el material no resulte familiar y no evoque imágenes.
- Resúmenes, antes, durante y después de bloques de información
- Cuadros sinópticos que organicen conceptos; mapas, gráficas o esquemas; representaciones espaciales del conocimiento que se está aprendiendo.

- Ayudas de memorización como acrónimos o acrósticos.
- Emplear ejemplos y no ejemplos de los conceptos a aprender, enfatizando semejanzas y diferencias.
- Organizadores anticipados que se refieren a una afirmación introductoria de una relación o un concepto de alto nivel. Su función es dar apoyo para la adquisición de nueva información¹⁸.

Estas estrategias se consideran para el diseño instruccional del proyecto CELLA, adaptándolas, por supuesto, a las necesidades.

7. MEDIOS INSTRUCCIONALES.

Los medios instruccionales son todos aquellos recursos que apoyarán el proceso de instrucción. Los medios disponibles son a nivel auditivo y a nivel visual. A nivel auditivo se puede contar con sonidos o música que expresa alguna información importante. A nivel visual se puede tener texto o imágenes; a su vez, las imágenes pueden ser fijas o con movimiento.

Las imágenes fijas ayudan a recordar el contenido de materiales, son indicios simbólicos que pueden aclarar un mensaje. Ejemplos de imágenes fijas son las fotografías, los dibujos, los diagramas o los cuadros. Las gráficas muestran relaciones numéricas que permiten entender el significado de grandes cantidades de datos, mientras que los cuadros o diagramas expresan en forma visual una idea o concepto difícil de entender en forma oral o escrita. Los diagramas ayudan a una

¹⁸ Woolfolk, A., Educational Psychology, 1987.

representación clara o característica específica de un objeto, concepto o fenómeno¹⁹.

En éste proyecto se considera de manera muy importante los medios audiovisuales, ya que uno de los objetivos es que el alumno pueda observar de la manera más real posible los procesos celulares.

B. EVALUACIÓN DEL APRENDIZAJE.

Desde el análisis de necesidades educativas, hasta el diseño de la estrategia instruccional, se ha exigido una definición más o menos precisa de lo que el alumno aprenderá. Y necesariamente habrá que idear una forma de constatar si el alumno aprendió realmente. Este es el tema de la evaluación. El punto específico que corresponde plantear aquí es el de planeación de la evaluación del aprendizaje. Siendo el aprendizaje el punto central que se persigue con un programa educativo cualquiera que sea su naturaleza, justo es que desde el principio se planee su evaluación.

La evaluación del aprendizaje es el acopio sistemático de datos cuantitativos y cualitativos que permiten determinar si los cambios propuestos en los objetivos de aprendizaje se realiza en los alumnos. La evaluación se diferencia de la medición, la que sólo determina el grado en que el estudiante posee una característica; interpretación que hace posible elaborar juicios de valor que permitan tomar decisiones²⁰.

¹⁹ Brown, W. et al, Instrucción audiovisual, 1981.

²⁰ Schytter, L. G., "Aspectos generales de la evaluación del rendimiento escolar", en F. García. Paquete de autoenseñanza de evaluación de aprovechamiento escolar.

Existen diversas concepciones sobre la evaluación en la educación, que también se dan en la evaluación del aprendizaje:

- Como juicio de expertos. El valor lo asigna la experiencia.
- Como medición. El valor se asigna con base en un parámetro
- Como comprobación de la congruencia entre resultados y objetivos. El valor se asigna al final del proceso.
- Como un enfoque sistemático. Evaluación inicial (de requisitos y de entrada), evaluación durante el proceso (principalmente formativa) y evaluación final (principalmente sumativa). Todas ellas permiten retroalimentar el proceso y a los sujetos que intervienen y hacer ajustes a diferentes niveles.

Este último enfoque es el más conveniente porque retoma diferentes momentos de la evaluación. La evaluación inicial se divide en dos, por un lado la de requisitos que permite corroborar si el alumno cumple con los conocimientos previos para iniciar el proceso planeado, y por otro la de entrada, que permite valorar que tanto sabe del tema. Durante el proceso, la información que arroja la evaluación permite comprobar si se están logrando los objetivos; si los procedimientos y actividades de enseñanza son adecuados; si los objetivos son los apropiados en extensión y profundidad y si el alumno puede proseguir con otras fases. Esta evaluación se denomina *formativa*. La evaluación final, además de ser un índice global que condensa los resultados de la actividad de enseñanza - aprendizaje, permite asignar a los alumnos una calificación que promoverá o no al alumno al siguiente nivel, por ello se denomina *sumativa*.

Para la evaluación del aprendizaje dentro de un diseño instruccional, la evaluación formativa es la más útil ya que permite:

- 1) una mejor aproximación al aprendizaje del alumno que es hacia donde se dirigen los esfuerzos,
- 2) da cuenta de los aspectos específicos que son útiles en la instrucción y de aquellos que fallan,
- 3) no sólo sirve para trabajar con el aprendizaje del alumno sino también para valorar el diseño, los materiales, el orden, etc.

Para la selección de las técnicas e instrumentos de evaluación se debe tomar en cuenta: su congruencia con el tipo de aprendizaje; que permitan la integración de conocimientos nuevos con base en los anteriores; que faciliten su reproducción y su revisión, que sean susceptibles de ser aplicados en cualquier momento; que sean elaborados y revisados por profesores que imparten el curso, y, que cubran la totalidad de los conceptos básicos revisados en el programa.

Es importante mencionar que si bien la evaluación del aprendizaje es un elemento esencial dentro de la elaboración del diseño instruccional, su instrumentación no debe incluirse necesariamente dentro del programa educativo por computadora que se vaya a producir. Si el objetivo de aprendizaje y el contenido específico manejado exige emplear una técnica de observación mediante el uso de registros anecdóticos, esto deberá realizarse "fuera" de la computadora.

EVALUACIÓN EN EL PROYECTO CELLA

Debido a que la meta de aprendizaje de este proyecto se refiere a un conocimiento de tipo declarativo, la evaluación consistirá en dos clases de evaluaciones, evaluaciones parciales entre la exposición del contenido al finalizar un tema, esta evaluación se efectuará a través de reactivos de opción múltiple; la evaluación final con cuestionarios de opción verdadero - falso y con calificación numérica.

En resumen, en este punto se presenta la propuesta sobre la elaboración del diseño instruccional con orientación cognoscitiva y dirigido al uso de la computadora como apoyo del aprendizaje. El diseño instruccional debe ser un trabajo comprometido, flexible y dinámico. Los elementos presentados deberán adaptarse al contenido particular que se maneje y a la meta de aprendizaje que se desea logren los estudiantes. Es importante que los ocho pasos de que consta el diseño instruccional se realicen cabalmente, independientemente de que queden incluidos o no dentro del software educativo, como es el caso de los objetivos de aprendizaje o evaluación. Hay que recordar que el uso educativo de la computadora es únicamente un apoyo más al proceso enseñanza - aprendizaje.

3.3 DISPOSITIVOS DE USO.

Este punto aborda la segunda etapa del desarrollo: los dispositivos para el uso. Consta de dos incisos²¹; en el primero de ellos "Dispositivos para el uso" se hace

²¹No se incluyen todos los Incisos de la metodología del CISE por no ser necesario citarlos como Información, sin embargo esta conjunto de conocimientos se utiliza para poder desarrollar el producto.

un análisis del equipo en el que se va a usar el producto terminado, desde las características individuales del equipo hasta las instalaciones en donde se colocarán y usarán; en el segundo "Sistemas de Autoría", se ofrece una introducción a las herramientas disponibles para el desarrollo de software educativo.

DISPOSITIVOS PARA EL USO

El desarrollo de cualquier uso específico de la computadora en la enseñanza es un proceso que requiere conocimiento, esfuerzo, recursos y tiempo. Por éste motivo es necesario medir los requerimientos y beneficios. La consideración se hace en términos de costo - beneficio, por lo cual tiene que considerarse lo siguiente:

- Con el desarrollo que se propone se ganará algo en aprendizaje que por el contrario, no se ganaría con la forma habitual de hacerlo.
- El desarrollo propuesto, tendrá una economía respecto a la forma habitual de hacerlo.

Otra consideración, al contar con recursos escasos limitados, es que debe considerarse la orientación de los esfuerzos, la aplicación del conocimiento y el gasto de los recursos. Por ello es conveniente plantear los siguientes interrogantes: ¿Se plantea el uso de la computadora como una solución o como la adhesión a una moda?, ¿Hay una forma alternativa más efectiva de lograr los mismos propósitos?

Objetivo.

Prefigurar los requerimientos para el uso y condiciones de distribución de los productos del proyecto. La generación de una aplicación de las computadoras al aprendizaje y a la enseñanza, es un ciclo que se inicia por la determinación de uso que va a tener.

Uso: Para el cumplimiento de los objetivos didácticos y educativos de un uso determinado de la computadora con fines docentes, es condición indispensable disponer del equipo de cómputo adecuado, de las instalaciones para alojar dicho equipo, y de un procedimiento que regule el acceso de los estudiantes a las instalaciones y al equipo. En su defecto, si se trata de un uso independiente, dar al estudiante facilidades necesarias para que pueda beneficiarse del producto o programa. Con frecuencia se ha considerado que estas condiciones se resuelven automáticamente, por sí mismas, y la consecuencia más directa de ello es que programas que han requerido conocimiento, aprendizaje, esfuerzo, recursos económicos y meses de trabajo, quedan inactivos con más frecuencia de lo imaginado, por falta de condiciones adecuadas para su uso. En este punto se desglosarán cuáles son las condiciones para el uso de los productos y programas.

Planeación: Una vez determinado el uso y caracterizado a los usuarios, se fijarán los límites de tiempo, las fases que tendrá el proyecto, las tareas individuales, los recursos que se usarán, el financiamiento del proyecto, etc.

Desarrollo: En esta fase se hace uso del conocimiento y se "aterrizan" las ideas. El eje de esta fase es la elaboración de la maqueta del proyecto. A ésta meta le denominaremos prototipo. La fase de desarrollo alimenta a la fase siguiente

con un plan de producción que incluye toda la producción intelectual relativa al proyecto.

Producción: Durante esta fase, los prototipos se convierten en productos materiales, impresos, diskettes, grabaciones de video o audio, que son finalmente los que llegaran al usuario.

Definición de la población objeto.

Identificación de los usuarios.

Generalmente se concibe el desarrollo de una solución alterna a un problema de aprendizaje o de enseñanza en relación con las necesidades de ciertos sujetos. Por ello, el primer paso es caracterizar a los sujetos a través de atributos de identificación (género, edad, nivel, escolar, localización geográfica, etc.), ésta caracterización sirve para establecer quien será el usuario del producto. Recuerdese que ya se había establecido anteriormente los usuarios a los que va dirigido este proyecto.

- Los usuarios son alumnos de la materia de Biología que cursan el segundo año en educación básica secundaria, con una edad promedio 13 años. Son grupos mixtos y sus conocimientos previos son de Ciencias Naturales en el nivel básico, y de la materia de Biología en el primer año.

Forma de Acceso.

La forma de acceso puede ser curricular o independiente. Como independiente se entiende cuando el producto se diseña para usarse sin la mediación de un sistema escolar y en este caso tienen que prever las condiciones en que será usado. La forma de acceso curricular es la que utiliza el proyecto CELLA, con esta forma el producto pasará a formar parte de las experiencias prescritas en un programa de enseñanza regular, por lo tanto debe circunscribirse a los límites de tiempo y espacio de la escuela o programa de estudios. El proyecto se apega al plan de estudios de la SEP²².

Quantificación de los usuarios potenciales.

Para plantear el uso educativo de la computadora es necesario expresar el número de usuarios potenciales. Para éste proyecto, recuerdese que el producto será usado como el software educativo del proyecto COEEBA. En el nivel secundaria se cuenta con los laboratorios de computación y las salas de estudio que es donde se usará el PCE (Programas de Computación Educativos) producido. Así que el número de usuarios se podría calcular como el total de los alumnos de segundo año que llevan la materia de biología, con la restricción de los equipos "disponibles"²³ en las instalaciones.

Equipamiento

Sobre éste las decisiones son difíciles, ya que existe una amplia variedad de marcas y características de equipos; además de la velocidad con que pueden ser obsoletos.

²²En el Diseño Instruccional, Inclsio 4. Análisis Instruccional.

²³Esta "disponibilidad" significa no el número de computadoras en el plantel, sino el número de equipos que funcionan adecuadamente para los usuarios.

En este proyecto se “evitará” de alguna manera éste problema ya que el desarrollo está orientado para el uso en equipos específicos, equipos que distribuye la SEP a sus planteles con plataforma²⁴ Intel y que en su mayoría son PC. Sin embargo cuando se desarrolla normalmente las decisiones sobre la plataforma y el equipo son críticas y debe ponerse especial cuidado en el costo y la obsolescencia determinados por el desarrollo tecnológico.

El seleccionar un software también es una decisión importante. Dentro del software debe considerarse primero el sistema operativo. Para éste proyecto dada la plataforma Intel, el sistema operativo que se “usará”²⁵ y también con el que se desarrollara el producto será el MS-DOS²⁶ utilizado por los equipos de la SEP. Además del sistema operativo existen otros tipos de software, a continuación se mencionan los más importantes de acuerdo al desarrollo de productos educativos:

- **Lenguajes de programación.** Son traductores de las instrucciones “humanas” al procesador que es quien las ejecuta. Existen lenguajes especializados para áreas determinadas. Este tipo de software requiere cierto grado de conocimiento para usarlos efectivamente, por lo tanto requiere de personal especializado para la programación. Lenguajes como C y Pascal se utilizan para desarrollar aplicaciones. El lenguaje que se utilizará para la programación del tutorial es Pascal. La razón de programar en éste lenguaje es que dado que a los

²⁴Las plataformas se clasifican de acuerdo al procesador, para J.M. Alvarez Manilla las plataformas en uso son: Familia Intel, Motorola y Risc.

²⁵Debe hacerse una distinción entre software para “uso” y para “desarrollo”. El primero es el que el producto necesita para operar y el segundo es el que se necesita para producir el producto.

²⁶Sistema operativo de MicroSoft. DOS (Disk Operating System) es el sistema operativo estándar para computadoras personales PC.

profesores de secundaria que tomaron cursos incluidos dentro del programa COEEBA se les enseñó Pascal como lenguaje de programación.

- *Sistemas de Autoría.* Están basados en los lenguajes de programación, pero se basan en “superinstrucciones” que facilitan al usuario la programación. Los más usados en el cómputo educativo son Hypercard para Macintosh, ToolBook para Windows, Authorware para Macintosh y Windows, Linkway para DOS, StoryBoard para DOS, etc. Es importante aquí mencionar que los equipos de la SEP cuentan básicamente con sistema operativo MS-DOS.

La adquisición de software generalmente incluye la licencia para usarlo en una sola computadora y está permitido hacer un número restringido de copias para respaldo. Cuando se desarrolla software para venta es importante tener la licencia ya que en muchos casos los fabricantes exigen el pago de regalías por cada ejemplar vendido.

El software para uso y para desarrollo tiene un costo económico, por esto debe estudiarse su vigencia. Se hace referencia a la vigencia de un material, incluidos los programas en dos dimensiones²⁷:

- 1) *Temporal.* Un material tiene vigencia temporal amplia cuando su temática, sus contenidos cognoscitivos o informativos son estables, es decir, no sufren cambios en largos periodos o son insignificantes. No es recomendable hacer producciones costosas de materiales cuyos contenidos tienen vigencia limitada.

²⁷Alvarez Manilla, CISE, 1994.

2) Espacial. La posibilidad de amortizar los costos de producción de un material está en proporción directa del número de usuarios potenciales. Se consideran dos tipos de amortizaciones: a) la económica, que consiste en recuperar monetariamente la inversión realizada en la producción, y ; b) la social, consiste en que la inversión que se realice, aún cuando no sea objeto de pago por parte del usuario final, sea productiva socialmente dado que beneficie a una población amplia, e incluso cuando no reporte beneficios económicos directos. En ambos casos, la posibilidad de amortización se verá restringida si los contenidos tienen una vigencia espacial localizada, o sea que sea de interés para una población muy circunscrita geográfica o socialmente.

La compatibilidad es otro de los problemas considerables al planear la forma de uso del proyecto. Dada la diversidad de estándares en equipo y de software, es difícil satisfacerlos. El desarrollador, al considerar la forma de uso deberá tomar en cuenta lo siguiente:

- 1) Si se plantea un uso no curricular del producto, es pertinente hacer un estudio preliminar para determinar los equipos de que disponen los usuarios potenciales.
- 2) En el caso de usos curriculares dentro de una institución o programa educativo, es conveniente desarrollar una norma con especificaciones del equipo y las configuraciones bajo las que puede operar el producto.

Diseño de espacios.

En cuanto a los espacios, el dónde y cómo colocar las computadoras por lo general es un problema para las autoridades escolares y los docentes. Cuando se considera el uso grupal, pueden agruparse orientadas a un punto en un salón, o en filas dentro del laboratorio; en usos independientes y curriculares que no requieren la intervención docente, el espacio ideal son las bibliotecas y los centros de recursos para el aprendizaje (instalaciones equipadas con dispositivos para el uso que permiten el acceso libre de estudiantes). De cualquier manera la colocación del equipo no debe ser arbitraria sino bien estudiada.

Administración.

La computación es un recurso caro, por lo cual, es conveniente plantear formas de uso que permitan una amortización apropiada de esos costos.

Hay que considerar riesgos durante el uso y desarrollo de software como son los virus y el robo. La mejor manera de prevenirlos es siempre hacer un respaldo de la información.

Un punto muy importante es la documentación:

Del desarrollo. Conforme el proyecto evoluciona es preciso producir documentos que ayuden en cualquiera de las etapas a entender lo que se planeo y lo que se ha realizado. También ayudarán a producir los manuales para el usuario que son requisito para que se haga un buen uso de los productos.

El definir las instalaciones, el equipo y la forma en que se usarán es muy importante. Lo mencionado anteriormente es lo mínimo que debe preverse para que el producto se use y no quede almacenado ya sea por que en el equipo que hay el software no corre, o porque no hay instalaciones adecuadas para usarlo, etc. Debe mencionarse que en este caso por obvias razones el costo no es muy importante, pero para otro desarrollo si lo es y mucho, por ello se incluyen las consideraciones en cuanto a costos.

SISTEMAS DE AUTORÍA.

Para el desarrollo de programas educativos se requiere, una vez hecho el análisis de la problemática educativa correspondiente y la elaboración del proyecto de desarrollo, elegir las herramientas más adecuadas para llevarla a cabo. Tales herramientas deben ofrecer las posibilidades de manejo de todos los recursos que estén planeados. Aunque la relación computadoras y educación se inició hace tiempo, en esos primeros tiempos los recursos utilizados eran pobres en calidad y cantidad, asimismo existían herramientas que integraran los diversos recursos de una manera sencilla.

En este punto se describen algunas de las diferentes herramientas que pueden ser usadas para la elaboración de programas educativos por computadora, se plantea que las idóneas son las que se denominan sistemas de autoría que son la integración, de las herramientas necesarias para desarrollar un programa educativo.

Sistemas de Autoría y su ambiente de trabajo.

Los sistemas de autoría son programas que proporcionan la integración de un conjunto de herramientas en un mismo ambiente de trabajo, además de ofrecer la posibilidad de un manejo estandarizado de todos ellos. La disposición de las herramientas en un ambiente de trabajo es el resultado de una evolución de los sistemas para el uso de la computadora y para el desarrollo de aplicaciones en ella.

Los programas establecen un ambiente de trabajo que puede ser: de línea de comando de texto, de comandos por menús o gráficos (más fáciles de usar). Si el sistema operativo tiene ambiente gráfico, las herramientas para construir programas aprovechan las facilidades gráficas del sistema.

Herramientas para la elaboración de programas.

La elaboración de software incluye varias etapas, una de ellas es la fase de desarrollo del paquete, en ella se utiliza algún software que permite llevar a cabo los planteamientos dados en la etapa del diseño. Por lo tanto, la herramienta del desarrollo debe realizar todas las opciones de diseño.

La herramienta de desarrollo puede ser de varios tipos:

- un lenguaje de programación de propósito general,
- un lenguaje de programación de propósito específico,
- un paquete específicamente diseñado para permitir el desarrollo de un área concreta.

Por otra parte, con la herramienta de desarrollo puede trabajarse con la computadora y el producto a partir de un ambiente integrado, o bien tiene que dividirse el trabajo en tres momentos mínimamente:

- momento de elaboración en computadora (se supone que ya se ha hecho un análisis y diseño y que el momento es de elaboración en una herramienta específica).
- momento de corrida y depuración (en este momento se hacen las pruebas convenientes que permiten terminar el producto).
- momento de ligado y compilación (en este se busca que el programa terminado pueda funcionar fuera del ambiente de desarrollo, es decir que sea ejecutable lo más independientemente posible).

Cuando una herramienta de desarrollo permite trabajar conjuntamente con la computadora, los dispositivos, y con el producto que se va elaborando , se considera que dicha herramienta tiene un ambiente integrado. De hecho, actualmente, la mayoría de las herramientas de desarrollo nos dan la oportunidad de trabajar en un ambiente integrado que permite dentro de él mismo llevar a cabo los tres momentos señalados anteriormente. Además, los ambientes permiten realizar otras tareas de importancia como la edición del programa (copiar, cortar, pegar) y definir librerías de funciones y rutinas. De esta manera el ambiente se hace más potente. Cuando alguna herramienta de desarrollo carece de ambiente propio es posible trabajar con un ambiente externo y simular un ambiente integrado. A continuación se presentan los tres tipos de herramientas mencionados anteriormente:

1. *Lenguajes de programación de propósito general.* La mayoría de lenguajes de programación presenta actualmente un ambiente integrado. En estos lenguajes puede desarrollarse cualquier tipo de programa y se clasifican generalmente por niveles; lenguajes de bajo nivel (ensamblador) y alto nivel (fortran, pascal, cobol). Todos han sido usados para el desarrollo de programas educativos, pero los más usados son los de alto nivel. Aunque son poderosos y casi sin limitaciones para hacer cualquier aplicación, estos lenguajes han mostrado requerir de bastante capacitación, lo cual desvía al productor de la realización del producto. De cualquier manera esto no debe representar ningún problema puesto que con un equipo multidisciplinario el experto en cómputo se ocupara de la programación.
2. *Lenguajes de programación de propósito específico.* Dado que los desarrolladores forman bibliotecas de subrutinas, funciones, etc. , se pensó en formar lenguajes de propósito específico que contengan las funciones estándar útiles para ciertas aplicaciones. En el caso de aplicaciones para la educación se desarrollaron lenguajes cuyo propósito era contener las funciones, rutinas y rutas más usadas en un programa educativo fuese tipo CAL (Computer Assisted Learning - AAC Aprendizaje Asistido por Computadora -), o tipo CAI (Computer Assisted Instruction - IAC Instrucción Asistida por Computadora -), un ejemplo importante de este tipo es el lenguaje PILOT. Estos lenguajes son denominados lenguajes de autor y requieren el conocimiento de la estructura de un programa aunque no en un alto grado. La labor de desarrollo de estos lenguajes ofrece otras facilidades: a) lo que el lenguaje no puede hacer se complementa con rutinas creadas en otros lenguajes (de propósito general), b) existen productos elaborados por terceros que se acoplan a las características del lenguaje e incorporan acciones que el lenguaje no posibilitaba directamente: manejo de gráfico animación, sonido, etc.

3. Paquetes diseñados para facilitar la labor de desarrollo de software educativo, denominados *Sistemas de Autoría*. La característica más sobresaliente de estos paquetes es que además de presentar un ambiente integrado (y actualmente con interfaz gráfica), proporcionan al usuario la facilidad de manejar opciones ya elaboradas y directamente accesibles para crear un programa educativo. Estas facilidades son manejables en lo básico sin requerir grandes conocimientos de programación. Basta con tener claro el diseño educativo, tener principios de desarrollo coherentes y aprender el funcionamiento básico para obtener un producto de buen nivel.

Cuando el usuario selecciona las opciones de trabajo el sistema autor va realizando una programación interna, generando el código de programa en el lenguaje propio del sistema. El producto desarrollado puede compilarse y ejecutarse sólo o junto con un módulo runtime. Estos sistemas tienen además la posibilidad de aprovechar rutinas y funciones elaboradas en otros lenguajes de programación, de manera que su potencialidad se amplía.

Sistemas de autoría para el desarrollo de programas educativos computarizados.

HYPER CARD.

Es uno de los sistemas de autoría más versátiles y poderosos para plataforma Macintosh. Permite incorporar imágenes, gráficas, audio, animación y vídeo, todo ello en un tarjetero que puede contener una cantidad considerable de tarjetas. Los tarjeteros HyperCard son el resultado final de un proceso de desarrollo, y son

capaces de contener todos los recursos de multimedia utilizados en su realización, así como las rutinas de programación, los elementos de diseño gráfico e imágenes necesarios para el desempeño de funciones educativas. El lenguaje de programación propio de Hyper Card se llama Hyper Talk y es muy sencillo, dado que Hyper Card esta formado por objetos, y que cada uno de ellos se puede programar de manera individual, todas las rutinas de programación que contiene el tarjetero mediante todos sus objetos se ejecutan en la medida en que el usuario las activa cuando está usando el tarjetero. Hyper Card está creado para el desarrollo de programas con hipertexto.

AUTHORWARE PROFESSIONAL.

Este sistema de autoría permite crear aplicaciones complejas en las que se incorporan gráficos, texto, animaciones, sonidos y vídeo de una manera sencilla y rápida. Aplicaciones complejas se refiere no sólo a manejar medios sino a que permite gran interactividad con el usuario y retroalimentación adecuada, además de una evaluación en cada una de las interacciones.

La programación se hace a través de iconos siguiendo una línea de flujo. El punto mas fuerte está en su capacidad de evaluar fácilmente el desempeño del usuario. También permite salidas del programa. Authorware se desarrolló primero para Macintosh aunque también existe en versión para DOS. Authorware esta creado para desarrollar programas con secuencias estructuradas y tiene gran capacidad en cuanto a formas de evaluación.

LINKWAY.

Es un sistema desarrollado por IBM que permite el desarrollo de programas educativos y de consulta de información de manera dinámica e interactiva. Linkway permite la integración, en un mismo programa de imágenes, animaciones y sonidos creados por otros programas; así como el control de dispositivos tales como vídeo disco, CD - ROM, etc. El atractivo de Linkway radica en su flexibilidad. Tres de las formas más relevantes en las que puede ser utilizado son : 1) como manejador de hipertexto o hipermedia; 2) como software de entrenamiento interactivo; y 3) como una herramienta de desarrollo de presentaciones.

ANIMATOR.

Este programa no es propiamente una herramienta de desarrollo pero combinado con una de ellas permite la elaboración y procesamiento de imágenes para crear con ellas animación como caricaturas, presentaciones, elaboración de material didáctico, etc. Los elementos anteriores no sólo son aplicables a dibujos sino también en tipografía. Animator es uno de los pocos paquetes de graficación en el cual se pueden incluir programas en lenguaje C; además de tener herramientas de captura y conversión de imágenes a diferentes formatos gráficos. El programa Aniplay es una opción para presentar las animaciones realizadas en Animator, donde se pueden controlar entre otros el número de veces que se repite la animación.

STORYBOARD LIVE.

Este producto es uno de los mejores en su categoría. Su estructuración y flexibilidad unidos a la posibilidad de utilizar diferentes medios (música, sonido,

vídeo, animación, etc.) hacen que sea una herramienta muy eficiente en el desarrollo de presentaciones, con color, animación y sobre todo, interactividad. Está desarrollado por IBM para la plataforma PC y para el sistema operativo MS-DOS . Dependiendo de las necesidades se pueden desarrollar presentaciones simples o presentaciones multimedia con animación, vídeo, música y efectos de sonido.

Fisonomía

Story board Live se compone de cinco módulos, cada uno de los cuales desarrolla una función específica dentro del objetivo general del programa de crear y presentar información. StoryBoard Live ofrece las herramientas de diseño necesarias en cada una de las fases de la creación y presentación de la información que se quiera mostrar. Basado en un diseño, al desarrollador o programador le será suficiente este paquete para poder realizar una "presentación" completa.

A continuación se hace una breve descripción de los objetivos de cada uno de los módulos componentes.

Picture Maker

Permite, como su nombre lo indica, crear imágenes o modificar las ya creadas o las capturadas (por otro de los módulos de StoryBoard, Picture Taker). Para crear una imagen Picture Maker dispone de diversas herramientas representadas mediante iconos y todos los elementos de dibujo llevan incorporado el microscopio que permite alcanzar de precisión en determinados detalles del diseño.

Picture Maker posee también un potente generador de gráficos de barra, de línea y de pie, que pueden incorporar de forma automática el resultado a un diseño. Picture Maker incluye librerías de imágenes. A las imágenes creadas con las diferentes herramientas se les puede incorporar textos gráficos que pueden ser tratados como otro gráfico. Es posible elegir entre varios tipos (fuentes) y estilos de texto (itálicas, subrayado, etc.); se puede resaltar texto mediante sombras, contornos subrayado, etc. Existe la posibilidad de determinar el espaciado entre caracteres y la justificación del texto en la línea que se edita. Picture Maker dispone también de un editor de tipos que permite crear tipos propios o modificaciones a los tipos estándar. En la edición de tipos se puede establecer la línea base sobre la que se desea asentar el tipo.

Picture Maker dispone de una paleta de colores que según el modo de resolución de la pantalla le permite incorporar hasta 256 colores. Como adición al diseño Picture Maker puede generar y llenar, de forma totalmente automática, las figuras creadas y en general cualquier área de diseño que este cerrada, con texturas estándar del propio StoryBoard Live , con texturas que el usuario puede generar o con texturas resultado de modificaciones de texturas existentes.

Picture Taker

Este modulo es la "cámara fotográfica" de StoryBoard Live . Permite capturar cualquier pantalla de imágenes o texto creadas por programas sobre DOS y guardarlas para su posterior inclusión en una historia o como parte de una imagen. Las imágenes capturadas se almacenan, varias o una sola, en un archivo. Según el

modo de almacenamiento de podrá o no hacer modificaciones sobre imágenes o pantallas de texto capturadas.

Story Editor

El objetivo principal de StoryBoard Live es realizar una "presentación" a través de una historia creada por medio de viñetas con una determinada continuidad. No se trata de crear imágenes o pantallas de texto individuales como puede realizarse con otro paquete para gráficos o con cualquier editor de texto; aunque esta posibilidad pueda utilizarse como un recurso mas de StoryBoard Live.

En este modulo es donde se programa el guión de la historia que se presentara al usuario. Con el Story Teller se programan (en orden) las imágenes en las historias, se tiene la opción de utilizar técnicas de disolución, fundido, explosión, etc.; entre imágenes y dentro de ellas.

Las imágenes se pueden crear de forma interactiva visualizándose de forma intermedia los resultados y efectos del "programa" de la historia. Las imágenes pueden visualizarse hacia adelante o hacia atrás y "saltarse" determinadas imágenes en un momento determinado de la historia, en función de opciones definidas por el programador. Se pueden incorporar a la historia determinados "sonidos" y "animaciones"; aunque para ello es necesario el hardware adecuado.

Cuando la historia es terminada puede almacenarse en cualquier disco y modificarse posteriormente para cambiar alguna secuencia o efecto del guión.

El guión se almacena en la Tabla de Historia, esta se compone fundamentalmente de los parámetros que determinan que es lo que se visualiza, durante cuanto tiempo, y como se efectúa la transición entre imágenes.

Story Teller

Este modulo es el que se encarga de “proyectar la película” o de visualizar la historia que se creó anteriormente con el Story Editor; la proyección puede hacerse mediante un monitor o un proyector de vídeo. Este modulo puede hacer de alguna manera que las historias sean “ejecutables”, es decir que no se necesite entrar al StoryBoard Live para correr la historia.

El desarrollo de programas educativos en computadora requiere de uso de herramientas que permitan su elaboración. Estas herramientas han ido evolucionando para hacerlas más poderosas y fáciles de usar, incluso se desarrollan para otras plataformas. Se mostraron y explicaron las herramientas más utilizadas en las diferentes plataformas. Pero como se mencionó anteriormente la herramienta que se utilizará para el desarrollo del proyecto es Story Board Live pues es la herramienta más poderosa en la plataforma DOS de acuerdo al objetivo que persigue el proyecto.

3.4 DESARROLLO DEL PROTOTIPO Y DEL PRODUCTO TERMINADO.

En este punto se trata la etapa llamada "desarrollo" en el modo general presentado; es la etapa en la que, a partir de la detección de las necesidades y con base en un diseño instruccional, se lleva a cabo la elaboración de los programas. Esta punto esta dividido en cuatro incisos en que se detalla el proceso de desarrollo de programas. En el primer inciso se presenta un panorama general que introduce los momentos y tareas del proceso; el segundo aborda de manera específica un aspecto crucial, el diseño de la "interfaz con el usuario"; el tercero ofrece una revisión de las herramientas disponibles para el desarrollo de programas educativos (herramientas de autoría); el último incluye elementos básicos del diseño gráfico, elemento fundamental para el éxito del software, y que involucra componentes como el uso de tipografía, color y la composición de pantallas. Se incluye en cada uno de estos incisos el desarrollo para el proyecto CELLA.

EL PROCESO DE DESARROLLO.

Las posibilidades educativas de la computadora han llevado a que un número cada día mayor de docentes decidan incorporarla en su práctica cotidiana. Esta decisión típicamente se convierte entonces en la decisión de crear u programa nuevo; es decir, tomar la opción de desarrollar nuevo software. Se ha señalado que esta decisión pudiera no ser la óptima, y que existen otras formas de integrar la computadora a la docencia como lo son la traducción y localización de software ya existente, la creación de experiencias de aprendizaje en torno a programas ya hechos, y el simple uso de la computadora como auxiliar en la presentación son formas mucho más fáciles y sencillas de incorporar esta tecnología a la educación que desarrollar nuevos productos.

No obstante, tal como se detalla más adelante, existen condiciones en que, en efecto, la decisión de desarrollar es la correcta. En ese caso, se requiere tener una idea, al menos general, del tamaño y complejidad del proyecto, a fin de no terminar el proceso con productos no terminados y recursos desperdiciados.

Se ha apreciado que muchos de los docentes que deciden desarrollar software no tienen una idea clara de lo que un proyecto implica. Por ello es pertinente el compartir ideas sobre el proceso de desarrollo, estas ideas son el resultado de malas experiencias y finalmente ayudaron a reconocer que es preferible enfrentar el desarrollo de manera sistemática y tan estructurada como sea posible. De esta manera, se abordan tres puntos, en el primero se define el proceso de desarrollo, en el segundo se describen las etapas principales del proceso y, en el tercero la composición interdisciplinaria del equipo de desarrollo.

- **Proceso de Desarrollo.**

Es el proceso de creación de software. Incluye no solamente la programación, sino la selección de contenidos, estrategias de uso, e incluso la documentación de los programas.

La programación no siempre es necesaria pues es posible producir programas sencillos usando solamente herramientas de autoría que "escriben el código" automáticamente; de hecho no se requiere un especialista pero si se necesita hacer la programación. Ejemplos de estas herramientas son Hyper Card y Authorware que se analizaron en el punto anterior.

Actualmente el desarrollo (en todas las ramas y particularmente en la comercial) es un proceso complejo; y en lo que toca al cómputo educativo, éste tiene que ser interdisciplinario si se busca eficiencia. Por lo tanto el desarrollo no lo pueden realizar sólo programadores ya que desarrollar implica la creación de una solución completa.

ETAPAS DEL PROCESO DE DESARROLLO.

El paso más importante del proceso es determinar si el desarrollo es necesario o no. Si el desarrollo nuevo no añade nada a lo ya existente; si tampoco aprovecha las oportunidades específicas que tiene la computadora como medio instruccional, entonces se tiene un caso más de lo que M. Álvarez Manilla²⁶ llama un "uso trivial de la computadora". Además esto sucederá acompañado de considerables inversiones de tiempo y recursos materiales y humanos que hubieran sido mas eficientes en otras tareas. En consecuencia, resulta crucial que la evaluación sobre la justificación del desarrollo sea resultado de una investigación y reflexión profunda.

Una vez determinado que el desarrollo se justifica, se inicia el proceso. Las etapas que se describen a continuación (y que a veces ocurren en paralelo, o en un orden no siempre lineal) son las normalmente reconocidas en el proceso, y son, en general, cuatro: diseño, instrumentación, prueba o depuración final, y entrega.

1. DISEÑO.

²⁶Ex - Director del CISE. UNAM.

Detección de necesidades, definición del objetivo del software.

Por principio debe asentarse en un documento cual es la necesidad del desarrollo, y cuales son los objetivos a cubrir. O sea, definir la tarea que el software permitirá realizar. En el caso de software educativo, esta definición dependerá de la particular detección de necesidades que se intenta resolver mediante el uso de la computadora, mismas que se plasman en los objetivos del diseño instruccional. Estos puntos, aplicados al proyecto CELLA están cubiertos en el análisis del problema educacional (esquema 3.2) y en el punto 5 del diseño instruccional.

Definición del usuario y del contexto.

No hay tarea sin usuario. El perfil del usuario es un elemento fundamental para guiar su desarrollo; por lo tanto, aquí se determinan sus características generales (edad, sexo, escolaridad, conocimientos previos sobre el contenido o tarea del software, familiaridad previa con la computadora, etc.)²⁹. Adicionalmente, debe definirse el contexto de uso, tanto en términos de los equipos y locales en que se utilizará el software, como del modelo de uso y las consecuencias de la tarea. Dado que este proyecto se realiza como un PCE³⁰ del proyecto COEEBA el equipo que se usará será el que la SEP ha distribuido en los planteles. Las características de estos equipos son las siguientes:

Computadora PC 286 (GAMA AT)

Memoria RAM 2 Mb

²⁹Esquema del Sistema de Producción. Esquema 3.3.

³⁰Programas de Computación Educativos

Unidad de disco 51/4 y 3/12

Teclado

Mouse

Monitor B/N y Color VGA

Disco duro 20 Mb

Impresoras de matriz de punto

Sistema Operativo MS-DOS

Además el software se desarrollará de acuerdo a la modalidad de un usuario por computadora y con la capacidad de correr en estos equipos sencillos.

Selección de herramientas de desarrollo.

Una vez que se determinó la tarea y el usuario, hay que decidir sobre la herramienta de desarrollo adecuada. En cómputo educativo se recurre normalmente a las herramientas de autoría; son menos poderosos los lenguajes de alto nivel, pero están optimizados para la elaboración de cursos, tutoriales y otros software educativo. Como se menciona anteriormente estas herramientas son para producir hipertexto, multimedia, simulación, etc. Y para todas las plataformas entre ellas se encuentran: HyperCard, Authorware y Chisel, para Macintosh; Toolbok, Linkway, Story Board Live y Animator, para MS-DOS y Windows. La herramienta que se seleccionó para el desarrollo fue Story Board Live dada su estructuración y flexibilidad unidos a la posibilidad de utilizar diferentes medios (música, sonido, vídeo, animación, etc.) además de su eficiencia en el desarrollo de presentaciones, con color, animación y sobre todo, interactividad; en el se pueden desarrollar presentaciones simples o presentaciones multimedia.

Es importante señalar que esta herramienta está incluida en la línea de especialización "Elaboración de programas educativos computarizados" impartida en DGSCA-UNAM y se considera la más completa y eficiente.

Selección de plataforma (desarrollo y entrega).

La plataforma es la combinación del procesador y el sistema operativo. Las más populares son Intel/ MS-DOS - Windows, Motorola/Macintosh, Motorola/Amiga. En la mayoría de los casos los desarrolladores, como es el caso de éste proyecto dado el programa para el que se desarrolla, pre determinan las posibilidades del proyecto a partir de una plataforma, de modo que la tecnología orienta la solución académica y no a la inversa.

Idealmente, es preferible especificar primero el objetivo y luego las herramientas que habrán de servir como plataforma. En realidad el proceso, por lo regular, es exactamente el inverso, dado que muchos desarrolladores adoptan de manera implícita lo que podría llamarse una estrategia "reactiva" y no de carácter "proactivo". En el primer caso, se desarrolla para aquellas máquinas que constituyen el mínimo común denominador existente (la base instalada). Esto asegura que el software desarrollado podrá correr en ésta base instalada y que el desarrollo será entonces de utilidad inmediata. Por desgracia, los ciclos de desarrollo suelen tener tiempos no triviales, por lo que esta estrategia reactiva puede tener como consecuencia la situación exactamente opuesta a la buscada: que para el momento en que el desarrollo se termina las máquinas en cuestión ya no son las más comunes, o de plano han sido descontinuadas. Por supuesto, dependiendo del tamaño y envergadura de los cambios tecnológicos, el software tal

ves podrá correr en las nuevas máquinas, pero desperdiciará seguramente parte del poderío ahora disponible. Por otro lado, si se sigue una estrategia totalmente pro - activa adoptando temporalmente tecnologías de punta el software no podrá ser usado porque las viejas máquinas disponibles no cuentan con los recursos suficientes. En consecuencia, para cada desarrollo debe hacerse un análisis cuidadoso sobre la mezcla necesaria de estas dos estrategias que conviene adoptar para el proyecto. Este análisis debe ser parte explícita del proceso de desarrollo, y no dejarse simplemente al azar de lo que en la institución patrocinadora ocurre en un momento dado.

Para este proyecto se considero el equipo de la institución, que es plataforma IBM compatible y utiliza el sistema operativo MS-DOS; pero el producto incluirá elementos multimedia que en el equipo actual no puede "usarse", pero si se actualiza, estos elementos funcionarán también en el nuevo hardware.

Elaboración de un mapa mental.

La idea de un "mapa mental" ha sido propuesta como herramienta para el desarrollo por Tina Van der Mollen, quien la ha utilizado con éxito en su programa para desarrolladores en Hyper Ed, de Nueva Zelanda (Van der Mollen, 1990). Es una especie de "lluvia de ideas" personal, en la que se vierte en una hoja de papel las características básicas que tendrá el software a desarrollar. Se inicia con una descripción muy breve del objetivo y el publico que se beneficiará con el software; luego se ubica, al centro, la idea rectora del programa, de la que se derivan, luego "ramas" temáticas. Este proceso puede continuar con la especificación de "subramas" de estas ramas principales, y así sucesivamente de manera

estructurada. Luego se añade a cada subrama la lista de tipos de material que emplearía (textos, gráficos, etc.), que pueden ser codificados mediante marcadores de colores. En seguida se señala que ramas se prestan para actividades interactivas; las fuentes de los materiales requeridos (cuando ya existan), o un a nota de que hay que desarrollarlos; finalmente, se ubican las ramas que se calcula serían las más fáciles y las más difíciles de desarrollar, para tener una idea de donde empezar y cuales serán las áreas más problemáticas. Este mapa, que debe elaborarse en un tiempo limitado, a fin de promover una mayor fluidez de ideas, se puede depurar y refinar de manera sucesiva. El mapa es una excelente ayuda para el paso siguiente, el de la pre - especificación, puesto que se puede regresar al mapa para calcular cuántos textos, gráficas, audio y vídeo se emplearán, que tipo de actividades requerirán de programación adicional, etc., lo que luego se puede traducir en elementos de estimación de costos y tiempos para el proyecto. El mapa mental del proyecto CELLA, después de las depuraciones, es el mapa conceptual Nivel 1, con sus respectivos subniveles³¹.

Elaboración de una primera especificación.

Se le llama "especificación" a la descripción escrita del resultado de las decisiones anteriores, más una visión general de cómo operará el programa y habrá de evaluarse su desempeño. Constituye la base para un entendimiento entre los participantes en el proyecto, así como el establecimiento del compromiso ante la institución patrocinadora o el cliente. Determina, en consecuencia, los objetivos del programa, el tipo de usuario y el contexto de uso esperado, la herramienta y la plataforma a emplear (incluyendo la especificación técnica al menos preliminar de

³¹Nivel1, Esquema 3.4; Nivel2, Esquemas del 3.5 al 3.11

los equipos), así como el plan general o estrategia de instrumentación que permitirá que los objetivos se cumplan. En ocasiones, si se incluye además un plan global o prototipo en papel, la especificación actúa como resultado y síntesis de la etapa de diseño. En términos prácticos, la especificación será la guía para poder determinar, cuando menos en forma preliminar, el costo y la duración del proceso de desarrollo, al apuntar hacia algunos de los insumos que se requerirán, tanto en equipo, software y accesorios como en recursos humanos necesarios. Es por ésta razón que la especificación resulta crucial, sobre todo en proyectos de desarrollo comercial, dado que permite facilitar tanto la realización del presupuesto como el compromiso ante el cliente, y dejar todo tan claramente explicitado como sea necesario. En ocasiones, una caracterización global basta; en otras, es indispensable definir parámetros. La especificación del proyecto CELLA se describe ampliamente en los puntos 3.2 Diseño Instruccional y 3.3 Dispositivos de uso.

Elaboración de prototipos.

"[...]La elaboración de prototipos es el proceso de diseñar algo, sea en la computadora o fuera de ella, que demuestre algunas de las características o elementos interactivos del sistema que se intenta construir. Al usar el prototipo se puede evaluar el diseño, incorporar cualquier cambio en el siguiente prototipo y, eventualmente, refinar el diseño para la aplicación o sistema final."³²

Para algunos expertos, este paso ya es en realidad un elemento de la etapa siguiente, la de instrumentación; otros insisten en que es el último e indispensable paso de la etapa de diseño. Lo cierto es que, a diferencia de las etapas anteriores,

³²Bauersfeld. Software by Design. 1994, p.155.

aquí se involucran cantidades importantes de recursos y trabajo, por lo que en algunos desarrollos comerciales la elaboración de prototipos es una tarea pagada (sea como pago final o como pago parcial del costo total de desarrollo). En cualquier caso el prototipo actuará como vehículo interno de comunicación, permitiendo que los diferentes miembros del equipo de desarrollo puedan saber que se espera de ellos y como se integran todas las partes; actuará también como una primera aproximación a las características de uso real del producto, al incluir un primer intento de la llamada "interfaz con el usuario", es decir, la forma en que el programa en un contexto real, la suma de interacciones que habrá de darse entre usuario y máquina. Esta fase es muy importante, pues si las primeras pruebas de los prototipos muestran que el programa es difícil de usar, o demasiado lento en desempeño, ello indicará que hay que revisar la especificación y probablemente realizar cambios, mismos que pudieran impactar tanto a los presupuestos como a los tiempos de desarrollo.

Los expertos consideran que hay dos tipos de prototipos que cumplen funciones ligeramente diferentes; en el caso de los prototipos "horizontales" se intenta dar una idea global de la funcionalidad entera del programa, aunque ninguna de las funciones opere todavía en detalle. En los prototipos "verticales", las funciones simplemente son listadas, y se toma alguna o un par de ellas cómo módulos representativos del desempeño del conjunto, y se desarrollan en profundidad.

Así, en un prototipo horizontal tal vez queden expresados los menús y submenús generales, con las opciones dentro de cada menú, y sea funcional solamente la navegación global, con pantallas "dibujadas" que den una idea de como se verá y operará cada función. En un prototipo vertical se prefiere tomar una opción de algún

menú y desarrollarla en profundidad a fin de mostrar su funcionalidad interna , que se espera sea una muestra representativa del conjunto de funciones restantes. Por supuesto, un prototipo más completo sería el que incorpora ambos enfoques y mostrara ejemplos tanto en la amplitud como en la profundidad del desarrollo.

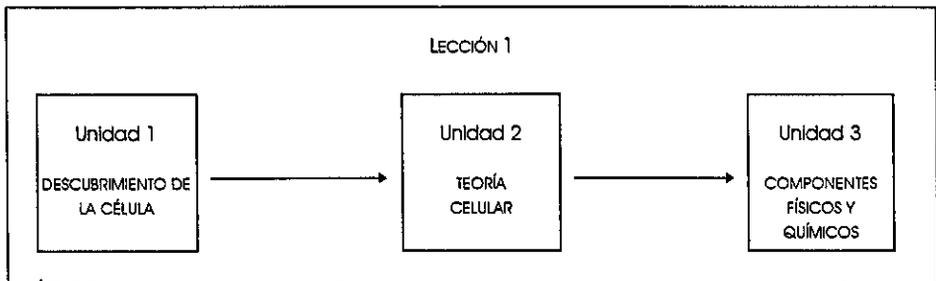
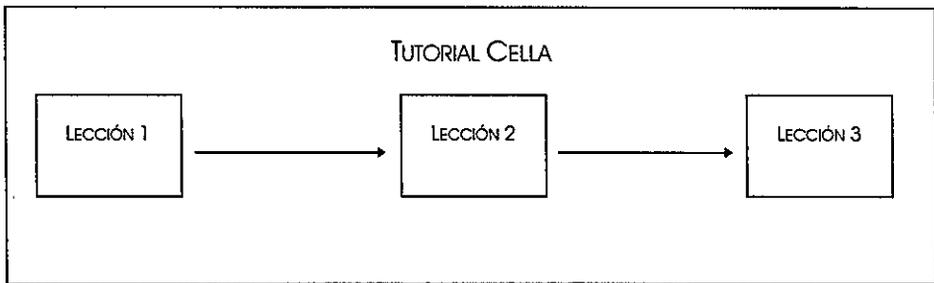
Para el proyecto cella se incluyen a continuación, un prototipo horizontal de las lecciones en que se divide el tutorial y las unidades que incluye cada una (esquema 3.12); y como prototipo vertical se incluye el fluxograma desarrollado en el curso "Introducción a la elaboración de programas educativos computarizados"³³. Para elaborar este prototipo primero se desarrolla el guión pedagógico que es un documento en el cual se especifican los conocimientos, habilidades o procesos que se desea transmitir o reforzar, así como la forma en que estos serán presentados al estudiante. Este debe contener:

- Contenido. El "guión" o texto que se va a exponer en el programa.
- Diseño de pantallas. Boceto que representa la distribución de imágenes, texto y color.
- Propuesta al usuario (interacción). Acciones ha tomar por el software de acuerdo a las respuestas o peticiones del estudiante
- Control de flujo. Indica la secuencia a seguir en el programa.
- Comunicación (usuario - computadora). Especificar la forma de interacción (iconos, botones, menús, etc.)

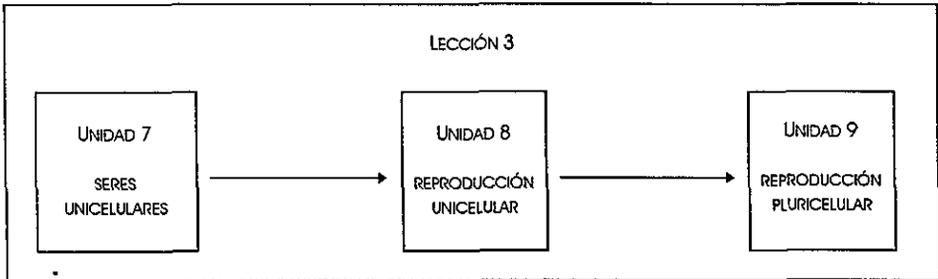
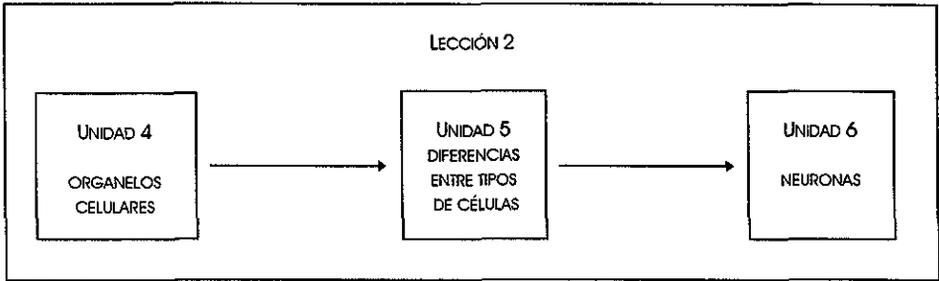
³³De la línea de especialización "Elaboración de programas educativos computarizados" impartida en DGSCA - UNAM.

- Resumen (muestreo - evaluación). Se debe incluir, junto con cada pantalla o conjunto de ellas, el objetivo de aprendizaje o una breve explicación del contenido de las pantallas.

A continuación se incluye el guión pedagógico de la unidad II (Esquema 3.13). Para su elaboración se utilizan formatos³⁴ que facilitan el desarrollo y el análisis.



³⁴Fuente: Lic. Semiramis Zaldivar G. Profesora de la línea de especialización y del curso "Introducción a la elaboración de programas educativos computarizados" DGSCA.



Esquema 3.12

GUIÓN PEDAGÓGICO

Tema: La célula / Teoría Celular (unidad II)

Objetivo: En la segunda unidad del tutorial el alumno memorizará los 3 postulados de la teoría celular. (Al finalizar la unidad II el alumno tendrá la capacidad de mencionar e identificar los 3 postulados).

Myrna H.
Septiembre, 1995.

TEMA: La célula / Unidad 2 / Teoría celular

OBJETIVO: Al finalizar la unidad II el alumno tendrá la capacidad de mencionar e identificar los 3 postulados.

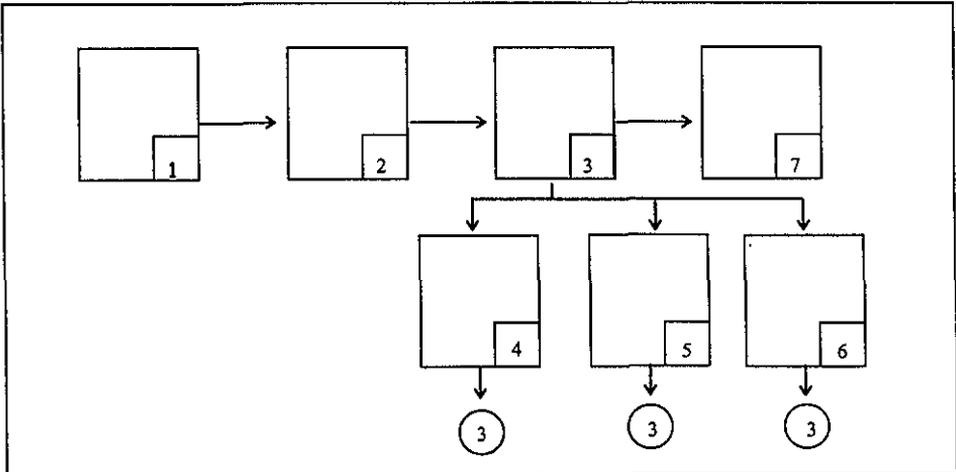


- Con animación se avanza el libro y se abre mostrando una hoja de postulados
- Pasa la página y muestro los 3 postulados

ORIGEN
 sólo pueden aparecer nuevas células por división de las preexistentes.

4

- Con la primera opción (pi) se desglosa la información de ORIGEN
- Con animación una célula se divide hasta llenar la pantalla



Esquema 3.13

Determinación final de requerimientos.

La elaboración de un prototipo global final dará una mejor idea de los requerimientos precisos del proyecto. Esto, a su vez, permite afinar calendarios y presupuestos, lo que puede implicar hacer ajustes con la institución patrocinadora o el cliente que financiará el desarrollo.

Dadas la características de cada desarrollo, se requiere una herramienta que cubra las necesidades del proyecto. Para el tutorial CELLA, se requiere, por ejemplo, una que permita mostrar gráficos de color, imágenes digitalizadas, visualizar las capturadas con scanner; una plataforma con buena velocidad de acceso, posibilidad de trabajar con imagen y texto combinados, capacidad de crear y manejar animaciones, muy buena interfaz con el usuario (uso de botones, iconos, etc.). A partir de éstas y otras consideraciones se decidió trabajar con Story Board Live. La plataforma y el equipo se deciden implícitamente por el programa COEEBA, plataforma DOS y máquinas con procesador 286 y monitor de 14".

En esta etapa es posible estimar cuanto tiempo se requiere para "scanear" y retocar imágenes, capturar y corregir los textos, diseñar los elementos de interacción, etc. Esto, a su vez, permite saber el tipo de personal involucrado, lo que multiplicado por la estimación del tiempo que cada uno aportará, da una idea del costo de recursos humanos y de los tiempos. Con esta información se puede realizar un cronograma y definir una ruta crítica. En el proyecto CELLA el desarrollo fue distribuido en tiempos irregulares, de manera que el producto final no se realizo

en un sólo intento, sino que a través de las presentaciones del prototipo y la conclusión del diplomado “Usos educativos de la computadora” y los cursos de la línea de especialización “elaboración de programas educativos computarizados” se fue mejorando el producto hasta llegar a esta última versión. De cualquier manera es importante mencionar la metodología (el “como debe hacerse”) ya que este texto puede servir de referencia a otros productores.

A un nivel de desarrollo simple, el prototipo ayuda a fijar los lineamientos de diseño de interfaz, diseño gráfico (tipografía, colores, etc.) y de codificación; a prever costos y tiempos; y establecer criterios de uso y de evaluación. Definitivamente, contar con una especificación y un prototipo constituyen la forma más eficaz de planificar y poder hacer el seguimiento de un proyecto.

2. INSTRUMENTACIÓN.

Esta etapa conocida también como la de “desarrollo” en el sentido estricto, consiste en transformar la especificación y prototipos finales en un producto probado listo para la entrega. Las actividades o fases importantes son:

Lineamiento de diseño y de uso de recursos.

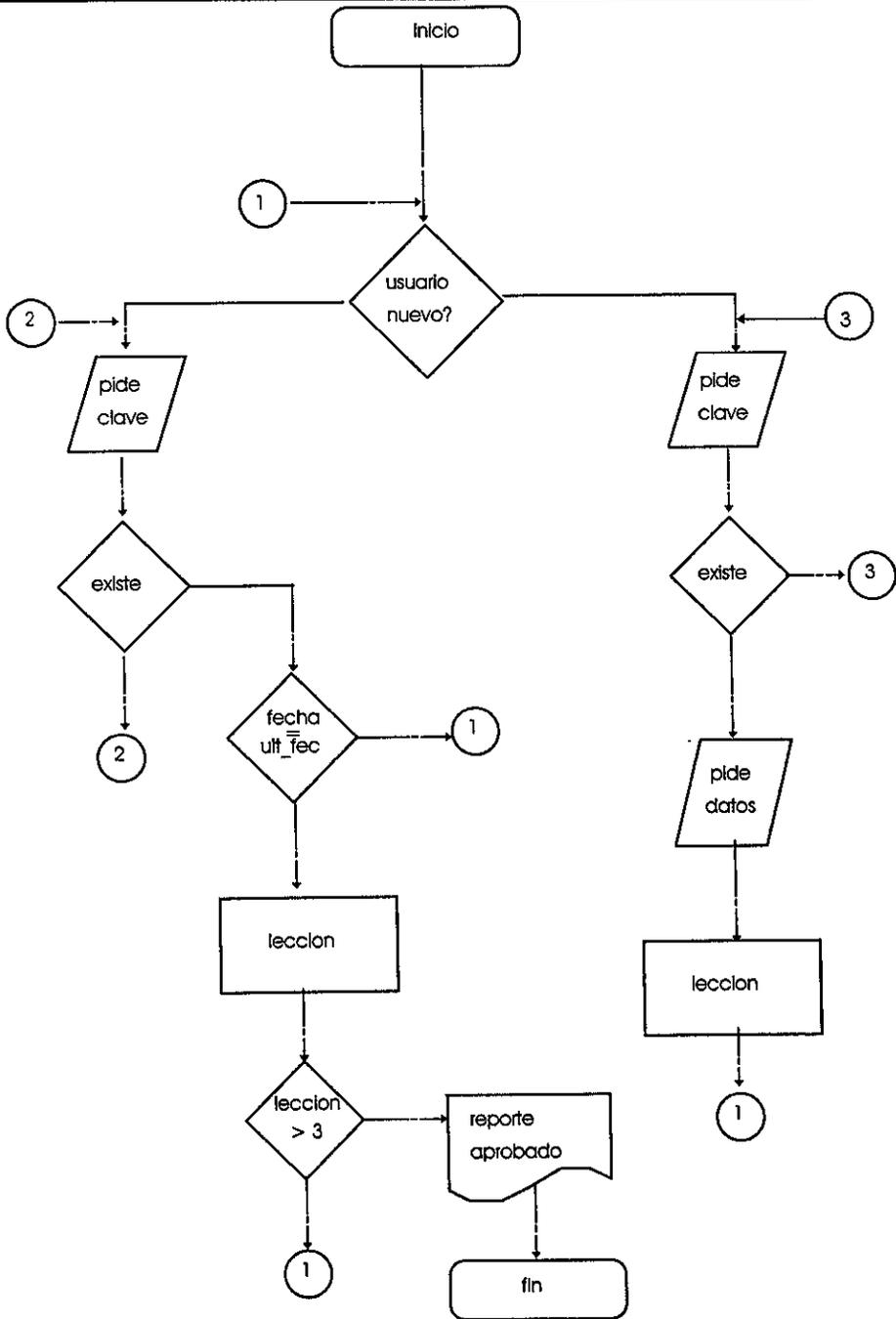
Consiste en elaborar un documento (tan detallado y formal como lo requiera el tamaño del proyecto) que oriente el trabajo de todos los participantes tanto en el

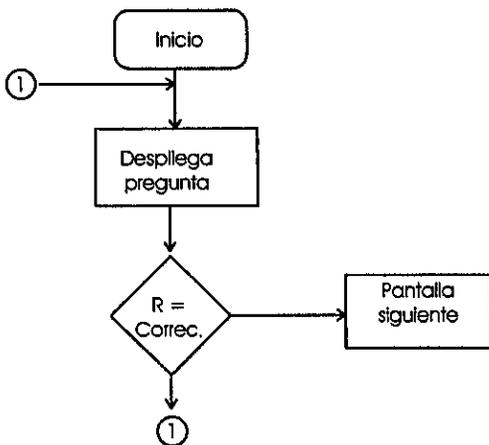
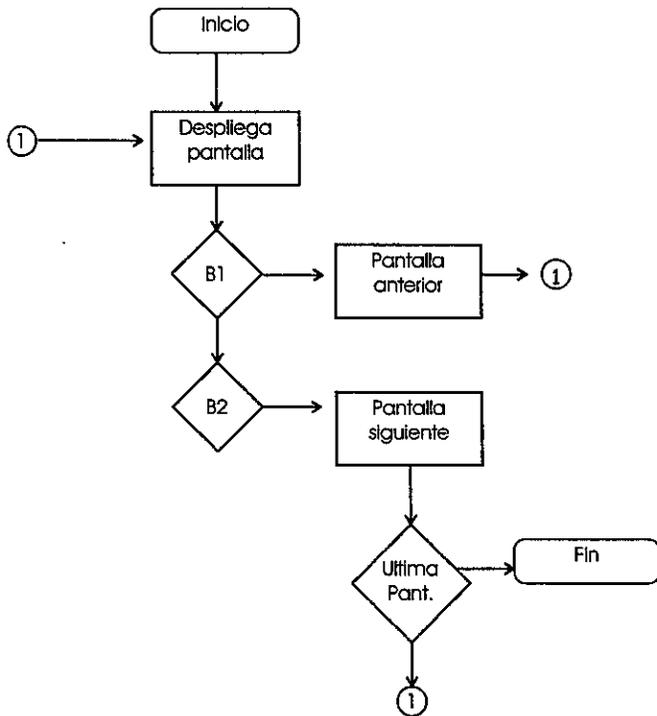
sentido de homogenizar el desarrollo (por ejemplo que todos utilicen las convenciones de tipografía y diseño acordadas en el prototipo) , como de programar el uso de recursos de desarrollo (scanners). En proyectos pequeños quizá esto no sea necesario, salvo por el control adecuado de la llamada "copia maestra", es decir, la ultima copia en la que se incorporan las versiones actualizadas de cada módulo del desarrollo; y por el control de los backups. Aunque el proyecto CELLA es de tipo multidisciplinario, el desarrollo, en su mayoría, fue individual contando con las asesorías necesarias. En cuanto a los recursos de desarrollo, en los prototipos se consiguieron muy pocos y por muy poco tiempo; la mayor parte de la digitalización se hizo durante el diplomado con el equipo del CISE.

Elaboración de pseudocódigo.

Es la descripción de los algoritmos centrales, los procesos a ejecutar, pero sin plasmar todavía en lenguajes de programación. Ya armado el pseudocódigo, éste facilita al programador convertir los algoritmos en instrucciones o comandos a algún lenguaje de programación. Si bien la traducción del código cambia de un lenguaje a otro, los algoritmos son transportables y reutilizables.

Para el tutorial CELLA se realizaron diagramas de flujo de los "procesos" principales. El primero es de el acceso al tutorial , el segundo es del flujo lineal a través del tutorial (en el tutorial también existe flujo ramificado) y el último es de la evaluación dentro del tutorial (preguntas en el transcurso del programa).





Elaboración de código.

El código es el conjunto de instrucciones o comandos que un programa ejecutará, expresados en algún lenguaje de programación. Esta es la parte que desarrolla el especialista en cómputo del equipo. Dependiendo del lenguaje, el código puede ser compilado para crear un programa "ejecutable"; la característica de los ejecutables es que pueden distribuirse y ser usados por los usuarios finales sin tener que incluir el ambiente, reduciéndose el tamaño y costo del programa. Cuando el lenguaje o la herramienta no producen ejecutables es preciso adquirir una licencia de uso. El tutorial CELLA se programó en Pascal; para el registro de usuarios, evaluación de cada lección e impresión de constancia; y en StoryBoard Live para el tutorial en sí (interacción, navegación, evaluación interna). En el apéndice A se muestran ejemplos de la programación externa en Pascal, y de la interna en StoryBoard Live.

Obtención / creación de materiales, creación de contenidos, obtención de derechos.

El código, a pesar de ser el alma del programa, normalmente es sólo un componente más en un proyecto de cómputo educativo, particularmente si se trata de un proyecto de multimedia. Normalmente, se requiere elaborar y depurar textos, plasmarlos en pantallas que es preciso diseñar y dibujar; nuevos diagramas, dibujos o animaciones creados en la computadora; o de imágenes fijas, vídeo o sonido que hay que digitalizar. Ello involucra la creación o la obtención de materiales, lo que a su vez implica procesos de investigación documental y gráfica, o producción de contenidos originales, tareas que consumen tiempo y recursos. En el caso de

utilización de materiales previamente disponibles, es importante recordar que normalmente estos materiales están protegidos con derechos de autor, lo cual puede involucrar el tener que obtener un permiso de derecho de uso, particularmente si el producto se comercializará. En muchas ocasiones, los autores o editores originales no requieren de pagos o licencias si el software será usado en educación y sin fines de lucro; en otros casos, si el uso es el equivalente a una cita o una referencia corta, entonces no se requiere, normalmente, más que asentar el crédito y fuente del material utilizado.

En cualquier caso, esta fase puede resultar, más compleja, difícil y tardada de lo que podría parecer inicialmente, por lo que debe incluirse un tiempo de margen considerable en la planificación del proyecto. El tutorial CELLA requirió digitalización de algunas imágenes, aunque aproximadamente el 90% de ellas se dibujo con mouse³⁵; las mayoría de las animaciones se hicieron por cuadro y otras basadas en los sprites de StoryBoard Live; los sonidos que se incluyen son de las librerías del propio StoryBoard Live y de otros programas. En cuanto al guión, como se mencionó en el punto de creación del prototipo vertical, se realizó con base en el plan de estudios y con fuente en la bibliografía existente para este nivel y para bachillerato. Algunas “pantallas” representativas del tutorial CELLA se encuentran en el apéndice B.

Integración de medios.

³⁵Estas imágenes se dibujaron a partir de las ilustraciones de bibliografía de diferentes niveles; diapositivas e incluso videos (para realizar las animaciones).

En esta fase se “arma” el programa, integrando el código, que actúa como estructura global, y los diferentes contenidos que habrá de incorporar el programa. Puede implicar tener que editar sonidos, imágenes y videos producidos en diferentes herramientas, en la herramienta de autoría o programación utilizada, lo cual puede implicar pasos previos de conversión de formatos. Las herramientas de autoría mas poderosas tiene la capacidad de facilitar esta integración al incorporar diferentes medios , tipos de archivos, e incluso permiten el control de dispositivos externos como videodisco, CD-ROM, CD, y dispositivos electromecánicos de diferentes tipos. Evidentemente, la integración de medios puede correr al parejo de la captura de nuevos materiales, de forma que estas dos actividades pueden hacerse parcialmente concurrentes.

La integración del tutorial se realizó en StoryBoard Live, esta es una herramienta muy poderosa ya que permite programar la presentación de pantallas, la “navegación” (lineal o ramificada), incorporar sonido e imágenes, además de permitir una muy buena interfaz con el usuario. El tipo de extensiones que maneja el Picture Maker son .PIC, por default, .GIF, .BMP, .PCX y TIFF; lo cual facilita la incorporación de imágenes extraídas de otro programa. StoryBoard cuenta también con un “capturador” que permite obtener imágenes “de la misma computadora”; y un controlador de audio y vídeo. En el apéndice A se encuentra un ejemplo de la programación en StoryBoard, al integrarse una lección del tutorial CELLA.

3. DEPURACIÓN Y PRUEBA PILOTO.

Depuración (debugging).

Una vez integrados el código y los contenidos del programa, hay que asegurarse de que éste corre de manera adecuada; o sea, tiene un buen desempeño, esto es, no es tan lento como para entorpecer la experiencia de uso; que es estable y no suspende la ejecución ante errores del usuario o problemas de comunicación con dispositivos periféricos; que es “correcto”, sin errores de codificación; que es “veraz”, sin errores en el contenido y, en el caso de software educativo, que es “eficaz”, en el sentido que permite que se cumplan los objetivos instruccionales planteados.

Para ello, se conducen primero pruebas internas, por el propio equipo de desarrollo, en lo que se conoce como “depuración”. Luego, es importante que personas no familiarizadas con el desarrollo evalúen, aunque sea de manera interna, en lo que se conoce como una “versión alfa”. Se logra así una “versión beta” completamente funcional y con todo el contenido final, que está lista para ser evaluada por terceros. Para el tutorial CELLA, la “versión alfa” se presentó en Expo - DIA 92³⁶, la “versión beta” se presentó en el Tercer Encuentro de Comunicación Educativa³⁷; el desarrollo de una nueva versión más enfocada el aspecto pedagógico se desarrolló durante el Diplomado “Usos Educativos de la Computadora”³⁸ y la última versión, producto final, es el resultado de los cursos de la línea de especialización “Elaboración de programas educativos computarizados”³⁹.

³⁶Desarrollos en Informática Avanzada, organizado por la FCA de la UNAM en noviembre de 1992.

³⁷Organizado por el Centro de Investigaciones y Servicios Educativos CISE, UNAM en abril de 1994.

³⁸Impartido en el Centro de Investigaciones y Servicios Educativos CISE, UNAM de agosto a octubre de 1994.

³⁹Impartidos en la Dirección General de Servicios de Cómputo Académico DGSCA, UNAM en 1995 y 1996.

Evaluación y ajustes finales.

Este proceso puede tener momentos sucesivos de evaluación - ajuste - nueva evaluación - nuevo ajuste, hasta que finalmente se logra una versión final, llamada de entrega o de release. La evaluación de terceros es importante antes de poder hacer la entrega final y equivale a lo que se conoce como "etapa de validación externa" en diseño instruccional. En el caso de software educativo es crucial garantizar que no persistan errores de contenido, proceso o eficacia instruccional. En el caso del software educativo es muy importante garantizar que no existan errores de contenido, proceso o eficacia instruccional. El autor muchas veces ya no percibe las deficiencias de su propio trabajo, a consecuencia de verlo repetidamente; la opinión de terceros es doblemente valiosa en ese caso.

La versión final deberá ser no solamente eficaz y correcta, sino amigable; esto se evalúa por referencia a la facilidad de aprendizaje, la retención de lo aprendido, el número de errores en la ejecución sucesiva por parte del usuario, y la experiencia subjetiva de uso. Todos estos criterios pueden evaluarse formal e informalmente mediante procesos de observación directa, por encuestas y cuestionarios estadísticamente significativos, mediante simulaciones de uso y "protocolos" de observación/ejecución a veces apoyados por dispositivos, como cámaras de vídeo o grabadoras de audio ante las cuales los usuarios comentan y describen lo que están haciendo. Independientemente del procedimiento, lo importante es reconocer que, la opinión del usuario es la que vale, independientemente de lo que el autor u otros "expertos" opinen.

PRODUCTO FINAL "TUTORIAL CELLA"

ESTRUCTURA.

TUTORIAL CELLA

I PRESENTACIÓN

- REGISTRO DE EL USUARIO EN EL ARCHIVO.
- INTRODUCCIÓN AL USO DEL TUTORIAL

II DESCUBRIMIENTO DE LA CÉLULA

- HISTORIA
- DEFINICIÓN

III TEORÍA CELULAR

- POSTULADOS
 - ORIGEN
 - UNIDAD
 - FISIOLÓGÍA

IV COMPONENTES FÍSICOS Y QUÍMICOS

- FÍSICOS
- QUÍMICOS

- MATERIA ORGÁNICA
- MATERIA INORGÁNICA

V ORGANELOS CELULARES

- NÚCLEO
- CITOPLASMA
- MEMBRANA PLASMÁTICA
- NUCLEOLO
- MITOCONDRIA
- RETÍCULO ENDOPLASMICO
- RIBOSOMAS
- LISOSOMAS
- CROMOSOMAS
- MEMBRANA NUCLEAR
- VACUOLA
- COMPLEJO DE GOLGI

VI DIFERENCIAS ENTRE CÉLULAS VEGETALES Y ANIMALES

- PARED CELULAR
- CLOROPLASTOS
- CENTRIOLO

VII NEURONAS

VIII SERES UNICELULARES

- TIPO VEGETAL
- TIPO ANIMAL

IX REPRODUCCIÓN UNICELULAR

- BIPARTICIÓN
- ESPORULACIÓN
- GEMACIÓN

X REPRODUCCIÓN PLURICELULAR

- CICLO CELULAR
- MITOSIS
- MEIOSIS

XI REPORTE DE APROBACIÓN

NOTAS DE LAS UNIDADES.

REGISTRO DEL USUARIO EN EL ARCHIVO.

Al inicio de la ejecución del tutorial CELLA, la primer pantalla después de la de presentación, pedirá al usuario su nombre y una clave que será su clave de acceso

en las dos siguientes sesiones, las cuales se controlaran por medio de la fecha para que el usuario no pueda tomar más de una lección por día.

El nombre del usuario y su clave quedaran guardados en el archivo del tutorial para controlar las sesiones, y al concluir estas se dará de baja en el archivo.

INTRODUCCIÓN AL USO DEL TUTORIAL.

Estas pantallas son para dar la "bienvenida" al usuario al tutorial y explicar la forma de navegar a través del tutorial, se puede manipular con teclado; o con interfaz gráfica a través de botones si se cuenta con mouse. Las evaluaciones "internas" pueden ser contestadas con el teclado, pulsando la letra de la opción; o dando click en el botón de la opción.

REPORTE DE APROBACIÓN.

Al finalizar el tutorial, el usuario recibe un diploma impreso de acreditación del curso, pues por la estructura del tutorial, el usuario no podría llegar al final del mismo si no hubiese tenido buenas evaluaciones (un promedio mínimo de ocho) en el transcurso del aprendizaje. Si el alumno, en la evaluación final de cada lección, no obtiene una calificación mínima de ocho deberá repetir la lección hasta que lo logre.

Al imprimir el reporte de acreditación el usuario es dado de baja del archivo de usuarios. Esto se considero en base a que se espera que el usuario que ha aprobado el curso, no deba repetirlo.

Este reporte incluirá el promedio de las tres evaluaciones realizadas (una por lección). El programa principal se desarrolló en lenguaje Pascal .

REQUERIMIENTOS.

Los requerimientos de equipo del tutorial CELLA son los mencionados durante el capítulo; recuerdese que el tutorial se desarrollo para un equipo específico, las computadoras de la SEP que se encuentran en las escuelas secundarias. Los requerimientos mínimos para ejecutar el programa son: Computadora 286, Hard Disk (9MB), Monitor VGA, Impresora (opcional), Mouse (opcional), RAM (1MB)

En cuanto al software se necesita únicamente el Sistema Operativo MS -DOS (versión 3.3 o superior) y el Story Teller (modulo de Storyboard Live). El programa principal es de tipo ejecutable y llama al Story Teller, de manera que lo único que se necesita para correr el tutorial es teclear el nombre del programa ejecutable. Para mayor referencia se incluye en el apéndice C un manual de instalación.

4. ENTREGA.

Esta es la última etapa del proceso, aunque no por ello significa que el desarrollo terminó, al menos para efectos prácticos⁴⁰. De hecho, involucra no sólo terminar el código e integración final, sino la documentación (manuales de instalación, del usuario, y en el caso del software educativo, las guías para el maestro y el usuario).

⁴⁰Pues de acuerdo con M. Gándara en su ley "gandarilana", "el desarrollo nunca se acaba; la versión entregada no es sino una nueva versión beta".

El trabajo no termina ahí, dado que hay que prever que los usuarios tienen preguntas tanto de la instalación como sobre el uso del programa; o detectan errores que hasta el momento no se habían detectado; o sugieren cambios y mejoras; o bien, ocurren cambios en el hardware que requieren a su vez cambios en el software. Todo ello cae dentro del rubro de "soporte" y puede llevar a la "actualización", esto es, a la elaboración de nuevas versiones, tanto para corregir errores como para mejorar o adecuar el programa a nuevas condiciones. En el apéndice C se incluyen un manual de instalación del tutorial CELLA.

Como se señaló anteriormente el programa principal se desarrolló en lenguaje Pascal debido a que no es posible un mantenimiento ni actualización permanente a las escuelas. Si se considera que los profesores que formaron parte del programa COEEBA dentro de su capacitación llevaron, como lenguaje de programación, Pascal; desarrollar el programa en el mismo lenguaje les dará, a estos profesores, la ventaja de poder personalizar, actualizar o complementar el tutorial por sí mismos.

EQUIPO INTERDISCIPLINARIO DE DESARROLLO.

Debe quedar claro que la simple multiplicidad de tareas y habilidades involucradas en un desarrollo de rango medio trasciende las posibilidades de un individuo aislado. Es indispensable contar con un equipo interdisciplinario, o al menos tener acceso a un grupo en donde el conjunto de las habilidades requeridas estén representadas. Como se menciona al principio de éste capítulo, quienes deben participar en éste equipo, en condiciones ideales, son:

- **Experto en contenido.** Alguien actualizado en su campo de conocimiento. El guión base fue desarrollado a partir de bibliografía del nivel educativo; el guión para el producto final fue revisado por el biólogo Carlos Bautista, profesor de la Facultad de Estudios Superiores Zaragoza UNAM y participante del grupo del diplomado “Usos educativos de la computadora”.
- **Experto en diseño instruccional.** Especialista en psicología educativa, , didáctica, pedagogía o similar. Que tenga experiencia o interés en el desarrollo de medios instruccionales. Esta asesoría estuvo a cargo de los profesores del diplomado “Usos educativos de la computadora” CISE; y de la profesora Semiramis Zaldivar de la línea de especialización “elaboración de programas educativos computarizados” DGSCA.
- **Experto en interfaz con el usuario.** Que entienda y tenga experiencia en facilitar la “usabilidad” de un programa, diseñando la interacción para que sea intuitiva y amigable. La asesoría fue dada por Manuel Gándara profesor del diplomado “Usos educativos de la computadora” CISE.
- **Programador.** Conocedor en lenguajes de computación , herramientas de autoría, etc. Esta tarea fue cubierta por la autora, tanto la programación “interna” como la “externa”.
- **Capturistas, digitalizadores.** Técnicos que manejen scanners y digitalizadores. Esta tarea fue cubierta por la autora con asesoría de técnicos del diplomado “Usos educativos de la computadora” CISE.
- **Diseñadores gráficos.** Expertos que produzcan materiales en la propia computadora o que supervisen la captura y digitalización de materiales. El diseño de las “pantallas”, así como algunos dibujos, estuvo a cargo de la D.G. Astrid Hernández.

- *Evaluadores (beta testers): expertos externos en las ramas mencionadas, que puedan opinar sobre el desarrollo; usuarios tipo, representativos de la población a la que se orienta el desarrollo, usuarios novatos, usuarios expertos, y evaluadores preferiblemente profesionales o con conocimientos de evaluación de software. Este punto estuvo cubierto desde la “versión alfa” ya que se presentó al público, la “versión beta” se presentó en ante desarrolladores, y la última versión fue presentada tanto en el diplomado “Usos educativos de la computadora” CISE, como en la línea de especialización “elaboración de programas educativos computarizados” DGSCA para que fuera “usada” por expertos.*
- *Coordinador. Persona con capacidad de organización, capaz de integrar y motivar a un equipo multidisciplinario, que pueda administrar recursos, conciliar agendas, cuidar calendarios de entrega, y en general ser el enlace entre los productores y patrocinadores del desarrollo. Esta tarea fue cubierta por la autora.*

Esto no significa que físicamente haya siempre éste número de personas en el equipo de desarrollo; en ocasiones algunos miembros del equipo pueden cubrir más de una función. Por otro lado, tampoco implica que todos estén incluidos en la nómina durante la duración total del proyecto. Algunos expertos pueden colaborar sólo en etapas relevantes. Lo que es imprescindible es contar con asesoría o trabajo continuo del experto en contenido y del programador, así como los capturistas que pueden llegar a cubrir el tiempo completo. Debe mencionarse que de los puntos anteriores, sólo se recibieron asesorías, por lo tanto “la producción” fue realizada por la autora.

COMENTARIOS.

El proceso de desarrollo es complejo pero divertido; ofrece la oportunidad de ejercer, al mismo tiempo, las capacidades creativas y las de planificación. Conviene considerar el proceso en su conjunto al momento de evaluar el grado de ambición o complejidad de un proyecto. Por otro lado es un proceso que requiere revisión y ajustes continuos que son necesarios para que los productos terminados satisfagan no sólo a los autores, sino a los usuarios que son los destinatarios de todo el esfuerzo.

No existe una sola y única metodología que garantice el éxito. Pero, es mejor tener una idea global de lo que se espera. Una de las recomendaciones producto de la experiencia de este desarrollo, es que la combinación de las dos metodologías usadas contempla todos los puntos importantes a considerar. Aún así, actualmente, se puede decir que cada autor desarrolla su propia metodología. Mas, como se menciono anteriormente, lo importante no es la metodología, sino el producto final.

3.4.2 INTERFAZ.

Se llama interfaz con el usuario a la forma en que ocurre la integración entre el usuario y la computadora⁴¹. Es esta interfaz⁴² la que condicionará el tipo de

⁴¹A diferencia de las interfaces de hardware, la interfaz que "conecta" al usuario con la máquina no es una tarjeta, sino la experiencia total de la interacción. Es todo lo que el

experiencia que el usuario tiene al interactuar. DE manera formal, según otros autores, “[...] la interfaz humana es la suma de todas las comunicaciones entre la computadora y el usuario. Es lo que presenta información al usuario y acepta la información del usuario.”(Apple Computer, 1987:xi).

Y es precisamente este factor. El de un diseño generalmente pobre en interfaz, el responsable de la mala reputación de las computadoras ser difíciles o frágiles. Por desgracia, lejos de entender que el problema está ahí, el usuario generalmente asume la culpa de una mala experiencia y se auto - descalifica o se da por vencido de antemano. Y cuando el usuario es un estudiante tratando de aprender algo nuevo o practicar materiales con los que tiene dificultad, resulta muy importante que la interfaz no lo desanime y acabe pensando que el problema es él.

Irónicamente, la interfaz es al mismo tiempo casi invisible al usuario normal (que piensa que las dificultades son deficiencia suya) y tan visible como para determinar el conjunto entero de la experiencia de interacción. De hecho:

usuario ve, manipula, oye y hace cuando se comunica con la computadora y ésta con él.

⁴²Existe un debate sobre la traducción correcta del Ingles “interface”. En rigor, aparentemente, debiera ser “interfaz” (algo que está entre dos caras). El término se usa también en geología y arqueología, en donde ha sido traducido indistintamente como “interfaz con el usuario” o bien “Interface al usuario”. SE ha adoptado “interfaz con el usuario”, gracias a las observaciones tanto del Dr. Alvarez Manilla (CISE), como del Dr. Hernández (Apple Latinoamérica), reforzados por un argumento producido en el diplomado “Usos educativos de la computadora”, si la interfaz es algo entre dos cosas, es necesariamente algo singular - el plural Interfaces al usuario no tendría sentido, dado que la interfaz es una. Y si es singular, el singular castizo correcto debe ser “interfaz” y no “interface”; al plural aplicable, tanto el uso genérico de “interfaz” con el usuario - como en Interfaces pueden ser de texto o gráficas - y a tarjetas de expansión y otros dispositivos físicos de comunicación entre la computadora y sus periféricos, puede entonces ser “Interfaces que la “z” es sustituida por la “c” de acuerdo al uso normal del español.

“La interfaz computadora/humano” es la superficie de interacción entre estos dos agentes. El diseño correcto de la interfaz al usuario debe poner el poder en manos del humano” (Mountjoy, 1990:XIII).

Esta cita pone de relieve el hecho de que, en efecto, la interfaz no es sólo un asunto técnico, sino que refleja la filosofía particular en cuanto al uso de la computadora: la interfaz determinará si el poder lo tiene la máquina o el usuario; si la operación debe facilitarse para el equipo o para el humano, según privilegie la facilidad y velocidad de ejecución para la computadora, o bien la calidad y disfrute de la experiencia por parte del usuario.

La evolución de la interfaz ha sido importante y se ha dado a través de los años, partiendo de una mala interfaz (sintaxis específica) hacia un ambiente gráfico. Los problemas de interfaz no sólo ocasionan mala reputación de algunas computadoras, sino que se traducen en pérdidas económicas. Estudios realizados durante los 80's mostraron que mientras menos intuitivos y fácil de usar era un sistema, mas se elevaban los costos de capacitación y el impacto de los errores de uso. De ahí que la “amigabilidad” adquiera importancia, ésta se puede evaluar por consideración de factores como el tiempo de aprendizaje, tasa de error en el uso, retención a largo plazo de operación del sistema, y, en general, de la satisfacción subjetiva que se experimenta al usar una computadora ⁴³. Por lo tanto la interfaz es crucial: la

⁴³Schneiderman, Ben. Design the User Interface. p 18.

facilidad o dificultad del uso dependerán, en buena medida, de un adecuado diseño de la interfaz al usuario.

IMPORTANCIA DE LA INTERFAZ EN LA EDUCACIÓN

En educación su importancia es doble, porque cuando se hace cómputo educativo de hecho se pide a los usuarios que desempeñen una doble tarea: aprender a usar el programa (y la computadora en sí) y, además aprender el contenido que es el objeto del programa educativo; si la interfaz no está bien desarrollada existe el riesgo de sobrecargar la capacidad de aprender del usuario. Si la interfaz es confusa, inconsistente, inestable y poco intuitiva, el usuario se pondrá cada vez más angustiado y frustrado condición que no lo conducirá a aprender el contenido presentado por la computadora.

DISEÑO DE LA INTERFAZ.

Un buen diseño de interfaz considera al menos cuatro elementos: el usuario, la tarea a realizar, el tipo de interacción y, en consecuencia, a los equipos y dispositivos para la interacción.

- El usuario. El primer precepto de diseño es "conoce a tu usuario" (Hansen, 1971⁴⁴). No todos los usuarios son iguales. Evidentemente, al diseñar software educativo será muy importante tener en mente el tipo de usuario para el que

está destinado el programa y considerar, cuando menos, la siguientes características:

Características generales. No es lo mismo diseñar para niños en edad pre - escolar que para pilotos. Los parámetros básicos de edad, grado de escolaridad promedio, estilos de aprendizaje predominantes e incluso factores culturales y de género, deben tomarse en cuenta para garantizar que la experiencia sea lo más eficaz y agradable posible.

Conocimiento previo. Este concepto se puede dividir en dos tipos: el "conocimiento de la herramienta", y el "conocimiento de la tarea" a realizar o el contenido a aprender. En cuanto al conocimiento de la herramienta, instara la experiencia previa en computo. Schneiderman ⁴⁵ clasifica a los usuarios en "novatos", "usuarios esporádicos conocedores" y usuarios frecuentes expertos. No obstante, aún un experto puede resultar ingenuo o poco sofisticado en el conocimiento que va a aprender mediante la computadora; y ala inversa un experto en física no es por ello automáticamente un buen usuario de programas de física (requiere un mínimo de conocimiento de la computadora como herramienta) ; y como Schneiderman señala hay casos en que el usuario puede ser nuevo tanto en el uso de la computadora como en el contenido que va a aprender . En este caso el diseño de la interfaz debe ser más cuidadoso.

⁴⁴*ibid.* p.66

⁴⁵*ibidem*

El contexto social y las consecuencias del uso. El impacto de error en el uso de un programa varía según las circunstancias. Es importante prever si el uso de la computadora afectará incluso la manera tradicional en que la tarea se realizaba. Hay que analizar los efectos que traerá la adopción de la tecnología sobre la tarea y sus agentes.

- La tarea. Hay que determinar que es lo que el usuario necesita hacer y cómo puede hacerlo mejor, con qué parámetros y criterios. En el caso de software educativo la tarea se convierte en el análisis instruccional del que se desprenderá un diseño de la instrucción óptimo para los objetivos de aprendizaje deseados. El criterio fundamental es que la computadora no es el centro: lo es la tarea, al usuario no le interesa la interacción en sí, sino la medida en que le permita (y no le obstaculice) realizar la tarea.
- Tipos o estilos de interacción. No toda la interacción es del mismo tipo. Tradicionalmente se daba mediante el uso del teclado donde el usuario escribía los comandos e instrucciones. La respuesta a su acción se daba en el monitor; la interacción era de tipo textual. Las dificultades clásicas de éste modelo (errores al escribir, dificultad para memorizar, etc.) llevaron a que se exploraran mecanismos en que las opciones estaban predeterminadas y el usuario simplemente las elegía de una lista (menú), o bien simplemente llenaba espacios en un formato (diálogos o fillins). El problema seguía siendo la dificultad para acceder a los menús y diálogos a través de secuencias de teclas o teclas de función, y aunque los formatos muchas veces requerían también de texto. Estas y otras consideraciones llevaron a un grupo a proponer una forma de interacción directa,

en la que el usuario manipula los objetos directamente en la pantalla. Este nuevo paradigma trajo consigo el uso de nuevos dispositivos (mouse, touch screen) y nuevos programas (programación orientada a objetos)⁴⁶. En este estilo de interacción el usuario “ver y señalar”, en vez de “memorizar y escribir”, dado que los objetos a manipular están presentes, siempre son de fácil acceso. La siguiente fase en los estilos de interacción, según muchos expertos, será la de interfaces de lenguajes naturales, y particularmente mediante el reconocimiento de voz y de escritura manuscrita.

- El equipo y los mecanismos de interacción.

Plataforma⁴⁷. Muchas de las aplicaciones educativas actuales adoptan un estilo de interacción no a partir de que dicho estilo sea el óptimo de acuerdo al análisis de tareas (análisis instruccional) realizado, sino porque el equipo con el que se cuenta no permite más. Las primeras PC tenían capacidad restringida de gráficos y sonido, por lo que sus interfaces eran interfaces de texto; la plataforma más popular depende del MS-DOS cuya interfaz es una interfaz de texto, por lo que si se quiere incorporar una interfaz gráfica, la mayoría de las veces se necesita Windows. Así, la primera consideración es la elección de plataforma (orientada a texto u orientada a gráficos y multimedia) ya que esta condicionará el tipo de interacción posible.

⁴⁶ La primera computadora en hacer uso comercial de esto fue la Macintosh, de Apple, en 1984. La primera Mac incorporaba ventanas, menús, mouse, sonido y gráficos de alta calidad.

⁴⁷Entiéndase como plataforma a una particular combinación entre el microprocesador y el sistema operativo. Las más importantes en microcomputadora son la “PC-compatible”, Macintosh y Amiga.

- *Mecanismos de interacción.* Se han señalado ya, algunos de los más frecuentes: el teclado (QWERTY, Dvorak o ergonómicos menos populares.), el mouse; otros un poco menos, el trackball, la pluma digital, la pantalla touch screen; la tablilla gráfica y los menos populares son los reconocedores de voz, el "guante" y el casco de realidad virtual (considerado como el siguiente paso en interfaces).

Proceso general de Diseño.

- *Criterios y lineamientos generales.*

Se señalan a continuación algunos de los más frecuentes, los cuatro primeros tomados de Norman^{4b}:

1. El usuario debe poder determinar fácilmente el estado en que está el sistema, así como las alternativas de acción. (Mensajes o visores de procesamiento)
2. Deberá contarse con un modelo conceptual consistente con la imagen global del sistema (como el escritorio de mac o de Windows 95).
3. La interfaz debe incluir un mapa de las relaciones entre los estados del sistema; en lo posible, el usuario deberá determinar que acciones tienen sentido en el contexto de acciones previas o subsecuentes, así como la secuencia en ciertas tareas que deben realizarse. (Elegir el tamaño de la hoja antes de imprimir).
4. El sistema deberá proporcionar retroalimentación continua al usuario. De nuevo, para cada acción del usuario deberá haber una reacción del sistema.

^{4b}Norman, Donald. The Design of Everyday Things. 1990.

Los siguientes criterios son algunos de los muchos que Apple recomienda, ofrecidos particularmente para la construcción de software en contextos educativos⁴⁹ :

5. El usuario deberá saber en todo momento dónde está, cómo llegó, a dónde puede ir desde ese punto. (ayudas de mapeo y navegación).

6. Deberá determinar que puede hacer y esperar de sus acciones; cómo corregir o cancelar acciones ejecutadas por error (reversibilidad de acciones), y en lo posible, el sistema deberá evitar que los errores puedan producirse.

7. Todos los teóricos de la interfaz están de acuerdo en que un criterio fundamental que debe cumplirse es la consistencia de la interfaz. Esta consistencia debe lograrse no solamente en el interior de un programa aislado, sino en el conjunto de programas y, preferiblemente, entre el diseño global de la interfaz del sistema⁵⁰ y la del programa específico.

ETAPAS GENERALES EN EL DISEÑO DE UNA INTERFAZ.

Se presentan a continuación las etapas generales del diseño de interfaz. El diseño de la interfaz para el tutorial CELLA se elaboró paralelamente al desarrollo del producto en sí (3.4 Desarrollo del prototipo y del producto):

⁴⁹Apple Computer, Inc. Human Interface Guidelines: The Apple Desktop Interface. 1987.

⁵⁰Un sistema es consistente si sus comandos o acciones son ordenadas, predecibles, describibles mediante unas cuantas reglas, y en consecuencia, fáciles de aprender y retener (adaptado de Schneiderman, Design in the User Interface, 1992).

- Definición de la tarea u objetivo que persigue el programa.
- Definición del usuario promedio, así como de los contextos de uso.
- Definición de la plataforma y los mecanismos de interacción (o adaptación a las condiciones existentes).
- Diseño general de los elementos de interacción (en el caso de interfaces gráficas, estos elementos incluyen típicamente botones, menús, iconos.), así como de los elementos de navegación y su distribución general en la pantalla.
- Elaboración de un primer prototipo, estableciendo de ser necesario, las transiciones entre diferentes momentos o módulos dentro del programa.
- Prueba del prototipo. Esta evaluación temprana es importante y puede realizarse incluso con recursos no digitales (guiones en papel, por ejemplo) o con funcionalidad simulada.
- Integración a la elaboración del código. Contando con un prototipo ya aceptable se puede proceder a la elaboración del programa en sí.
- Evaluación de la versión preliminar. Existen procedimientos estandarizados ("pruebas de usabilidad") que incluyen tanto procedimientos experimentales como de observación etnológica (grabando o entrevistando a los usuarios experimentales).
- Integración de la versión de la distribución. El que el programa finalmente se distribuya no significa que haya terminado el proceso de evaluación. Debe preverse tanto el soporte a los usuarios como mecanismos para obtener retroalimentación.

En el caso de software educativo, el área de evaluación de la efectividad instruccional es una de las difíciles, dada la relativa novedad del medio y porque promueven aprendizajes complejos y de tipos no muy frecuentemente evaluados en educación. Lo cierto es que la evaluación de la interfaz es un elemento importante dentro del conjunto de parámetros utilizados para evaluar el software educativo y su eficacia.

3.4.3 DISEÑO GRÁFICO.

El Diseño Gráfico juega un papel muy importante en la elaboración de programas educativos por computadora, ya que la estética y la funcionalidad colaboran para facilitar el aprendizaje y transformarlo en una experiencia agradable que invite a continuar.

Las bases de Diseño Gráfico como composición, color y tipografía, así como una metodología son muy importantes para la producción de un buen producto. A continuación se señalan los puntos más importantes de éstas bases:

COMPOSICIÓN.

El primer paso en el diseño con el que se trabajará. En este caso se trata de la pantalla de la computadora que normalmente tiene una dimensión de 640 x 480

pixeles. Una vez determinado el formato, hay que pensar en la composición, es decir, la combinación de elementos para obtener una síntesis armónica, unitaria y bella. Para este contexto, de acuerdo a las necesidades, se consideran como elementos compositivos : tipografía, color, gráficos, constantes de diseño y zonas interactivas. Para combinar estos elementos, de manera que se conserve la unidad a lo largo del programa, se utiliza una diagramación, es decir, el soporte de la composición.

EQUILIBRIO.

El equilibrio no sólo significa simetría e igualdad, ya que se puede encontrar equilibrio con pesos distintos si se toma en cuenta factores como la posición y el tono. Tipos de equilibrio: Equilibrio simétrico, en pesos distintos que se equivalen por la cantidad, logrado por la posición y logrado por el tono de dos pesos distintos.

CONTRASTE.

El contraste permite crear puntos de interés o de atracción visual; tiene dos niveles que se encuentran definidos de acuerdo al contexto cultural: el contraste mínimo y el contraste máximo. En el contexto occidental se aprende a leer de izquierda a derecha y de arriba a abajo, de ahí la preferencia por el lado izquierdo y el lado inferior del campo visual, por el desplazamiento de la vista de arriba hacia abajo hasta quedar apoyada en una base; así encontramos un esquema en el cual el ojo favorece el ángulo inferior izquierdo, este primer esquema nos da el contraste

mínimo. El contraste máximo se logra utilizando todo lo contrario el contexto cultural, es decir, el ángulo superior derecho.

Al diseñar pantallas de programas educativos se puede elegir ya sea el equilibrio o el contraste, o ambos si se quiere dar énfasis en alguna pantalla; no hay que olvidar la diagramación porque ésta conserva la constancia que a su vez dará consistencia al programa.

Al decidir la diagramación de manera general, se puede empezar a trabajar en cada pantalla individualmente, ya que cada una tiene sus propias características de acuerdo a su contenido. Este es un ejemplo para diseñar pantallas:

1. Enlistar elementos: Tipografía (bloques de texto y títulos), zonas interactivas (botones) y constantes de diseño (ornamentos), gráficos (vídeo, fotografía e imágenes) y, color y texturas.
2. Elegir una diagramación de acuerdo a las necesidades.
3. Designar el lugar para las zonas interactivas. Y las constantes de diseño.
4. Designar un lugar para texto y gráficos.
5. Elegir el color y/o textura que se utilizará.

COLOR.

Los colores pueden ser una herramienta que ayude visualmente a comunicar un mensaje, por eso es conveniente conocer ciertas reglas para utilizarla. Dado que la percepción del color es la parte más emotiva del proceso visual, tiene una gran fuerza y puede emplearse para expresar y reforzar una imagen visual.

El buen uso del color sirve al diseño para dar un mensaje de una manera eficaz, en este caso, es importante para establecer jerarquización, destacar y diferenciar elementos que tengan una particular importancia en las pantallas del programa.

Énfasis e impacto. Para obtener énfasis e impacto se deben tomar en cuenta cuatro factores:

- **Atención.** El color en algunos elementos ayuda a dar énfasis a los puntos donde se requiere atención, de éste modo el usuario puede visualizar rápidamente que es lo más importante.
- **Amenidad.** El color puede ayudar dando a las pantallas vistosidad, variedad y atractivo.
- **Claridad.** El dar color a algunos elementos puede ayudar a su entendimiento. Es necesario tener cuidado al diseñar con color porque el exceso puede ser contraproducente.
- **Retención.** El color en el diseño gráfico puede ser una eficaz ayuda, sobre todo si es importante que el usuario tenga ciertos elementos que van a servir a lo largo del programa. La retención puede ser lograda destacando algunos elementos de otros , o agregando cierto color característico y distintivo del elemento.

En general la utilización correcta del color en las pantallas, añade variedad, continuidad y renueva el interés del usuario⁵¹.

Como diseñar con color. Para diseñar con color hay que tomar en cuenta la relación entre los colores: primarios, secundarios y suplementarios. La rueda de los colores es muy útil para identificar las combinaciones que armonizan y aquellas que deben evitarse debido al gran contraste que provocan a la vista del usuario. La selección de dos colores complementarios en la rueda del color no es recomendable porque distrae la vista debido a que los colores parecen "vibrar" cuando están juntos. La selección de dos colores adyacentes o suplementarios crean armonía. La selección de matices con tres colores de por medio en la rueda de color crea contraste. Para obtener combinaciones de colores que contrasten, es recomendable usar colores oscuros en el fondo y colores más brillantes en primer plano.

Psicología del color. El color produce sensaciones y estímulos al hombre y a los animales por las diferentes longitudes de onda que proyecta. Este hecho se ha tratado de explicar atribuyendo al color efectos psicológicos, debido a que dentro de la psicología se estudian experiencias comunes que produce el color, como influencia en el estado de ánimo y en los sentimientos de los sujetos. En estos estudios se han establecidos diferentes modalidades del color:

- La frialdad o calidez de un color depende de la proximidad que este tenga al azul o el rojo respectivamente.
- Estados de ánimo de acuerdo al color:

⁵¹Hewlett Packard. Uso del color. 1993.

Colores tristes: neutros y oscuros (negro, azul oscuro, violeta)

Colores alegres: brillantes y claros (amarillo, rojo, verde)

Colores pesados: son los oscuros y saturados

Colores ligeros: son los claros y diluidos

La significación social y cultural común de los colores se debe tomar en cuenta ya que un mensaje con color podría cambiar un significado o remitir a otra cosa dependiendo del contexto del usuario. El color permite organizar el trabajo centrando la atención en lo más importante. Además motiva al usuario. Y dado que la motivación es un aspecto muy importante en el aprendizaje es recomendable tomarlo en cuenta.

Como usar el color.

- Use el color para explicar no como decoración
- Use el color sólo para establecer normas y fijar expectativas
- Coloque la información más importante en los colores más brillantes
- Use color para agrupar elementos o conceptos similares y establecer correlaciones (en este caso es conveniente utilizar pocos colores
- Use color para diferenciar elementos
- Use colores que contrasten para destacar los elementos principales y facilitar su lectura o visibilidad

- Use menos color (no más) Al limitar el uso del color se maximiza su impacto.⁵²

TIPOGRAFÍA.

El texto es el elemento o zona de comunicación que se usa casi siempre en las pantallas. La función del texto cambia según la estrategia que se desea seguir, puede ser usado para explicar una idea, para organizar ideas o funciones. Cualquiera que sea su objetivo debe usarse basándose en la estética y la efectividad. Se deben considerar varios factores:

- Es importante establecer que la caja de texto es un elemento en la pantalla, que tiene características que pueden tomarse en cuenta para la composición. Por ejemplo su tamaño, color y función.
- Es importante definir un estilo para mantener una consistencia en las cajas de textos y el diseño de la pantalla (el mismo tipo en los títulos, en las líneas de texto, justificación, interlineado, color, tamaño, estilo).

Una buena tipografía es aquella que nos facilita su lectura ya que entre sus características se encuentran: la visibilidad (claridad, buena definición e identificación) y la legibilidad (la clase de tipos que son adecuados para textos

⁵²Fuente: Hewlett Packard. Uso del color. 1993.

largos). Es importante verificar como se ve un tipo en la pantalla de la computadora.

Sugerencias tipográficas.

- Es recomendable usar sólo dos tipos de letra en nuestras pantallas (uno para títulos y otro para cajas de texto).
- Los tipos alargados son fáciles de leer en la pantalla
- El tamaño de las letras será determinado por la buena legibilidad en la pantalla
- El texto en mayúsculas no es muy legible
- Algunos manuales recomiendan las fuentes: Geneva, Avant Garde, Palatino o Bookman para títulos, y Nex Century Schoolbook, Geneva, Palatino o Bookman para bloques de texto
- Las cajas de texto deben ser justificadas, ya sea izquierda o derecha, ya que la mirada atiende un punto de retorno al que se acostumbra rápidamente
- Tratar de que el texto no de la apariencia de estar demasiado apretado, ya que eso lo hace denso y difícil de leer. Es importante lograr que el texto sea conciso cortando las palabras que no sean necesarias. Se recomienda que la caja de texto tenga siete palabras por siete renglones

- El interlineado debe permitir una buena legibilidad del texto. Para obtener un buen interlineado, éste debe ser de 2 a 4 puntos más grande que el tamaño de la letra.

El color en la tipografía.

- Debe ser utilizado sólo si es absolutamente necesario
- Puede ser un auxiliar para destacar, dar consistencia ya sea por razones motivacionales o estéticas.

METODOLOGÍA DEL DISEÑO.

El enfrentar un problema de diseño es algo muy difícil; a continuación se presentan algunos puntos que ayudan a seguir un proceso para resolver problemas de diseño de pantallas de un programa educativo. El diseño gráfico se basa en la investigación del contexto al que pertenece el usuario para utilizarlo como guía para encontrar la forma más eficaz de transmitir un mensaje. Los puntos más importantes para resolver un problema de diseño son:

1. Descomposición analítica del problema
2. Articulación y síntesis de los elementos o variables del problema

Primero, es importante listar todos los elementos y variables que intervendrán (ilustraciones, fotografías, iconos o elementos de interfaz, tipografía, animación y

vídeo), segundo, es necesario establecer niveles jerárquicos entre los elementos y definir su función dentro de la pantalla y con los demás elementos (explicar si una fotografía hará referencia al texto o si el texto da información sobre la fotografía). Es importante realizar diferentes combinaciones en la distribución, tamaño, color de los elementos para obtener opciones que nos permitan decidir cual es la más adecuada de acuerdo al propósito.

Es importante mencionar que el diseño gráfico de un programa educativo requiere un estudio de los usuarios y de su contexto, además de los elementos de composición y sus elementos; todo ello sumado a la creatividad y experiencia. Por esto es importante la participación de u profesional en el equipo multidisciplinario de desarrollo. Para la producción del tutorial CELLA se contó, como lo mencionamos en la definición del equipo de trabajo, con la asesoría de un Diseñador Gráfico. En el apéndice B se encuentran algunas de las pantallas representativas del tutorial.

3.5 EVALUACIÓN.

Este punto trata de la fase final de la elaboración de un material educativo por computadora. En la evaluación de prototipos por juicio de expertos, se hace una revisión de la validación que debe realizarse desde el punto de vista de diferentes especialistas que colaboran en el proceso de desarrollo: el especialista de diseño instruccional, en cómputo, en diseño de interfaz al usuario y en diseño gráfico.

EVALUACIÓN DE PROTOTIPOS A JUICIO DE LOS EXPERTOS.

Desarrollar software educativo requiere considerar muchos elementos y dedicar mucho tiempo. Un desarrollo a nivel prototipo funcional no estará completo hasta que no sea sometido a revisión por los especialistas: intervienen expertos en contenido, en diseño instruccional, en cómputo, en diseño de interfaz al usuario. A continuación se presenta un formato de evaluación de un prototipo educativo, realizado por expertos⁵³.

EVALUACIÓN DE PROTOTIPOS.

Una vez concluido el prototipo del material educativo propuesto, la primera evaluación deberá realizarse estimando que fuentes, instrumentos, procedimientos de recolección de información y reportes finales (incluyendo sugerencias de modificaciones) se deberán llevar a cabo. Por fuentes se entenderán los distintos expertos que participaron, aquellos encargados de validar, desde su perspectiva, la calidad del material en cuestión.

Los instrumentos de evaluación deberán prepararse a partir de los indicadores que cada experto considere dignos de tomarse en cuenta; tendrán que ser sencillos y lo suficientemente claros para permitir recolectar información útil.

El procedimiento de recolección de información deberá considerar el procesamiento y análisis de los datos que arrojen los instrumentos. A este nivel de evaluación no se deben considerar demasiado las cuestiones las cuestiones de carácter

⁵³Bañuelos Ana M. CISE.UNAM

estadístico. El conocer las características de los instrumentos de evaluación bastará para categorizar los datos descriptivos reunidos.

El último paso de la evaluación interna de prototipos terminados será la formulación de sugerencias sobre lo validado, que modificaciones o ajustes deberá hacerse en función del estándar de calidad que se desea⁵⁴.

En cuanto a formatos de validación por expertos, la literatura es escasa, igual que de EAC en general. Ha sido muy poco investigada esta primera etapa de evaluación de materiales, así como la necesidad de elaborar y probar formatos de validación propios. Entre algunos formatos se encuentra el de Galvis⁵⁵, quien propone un instrumento de evaluación basado en una escala que va de Totalmente de acuerdo a totalmente en desacuerdo. En dicho instrumento se toma en cuenta al experto en contenido, metodología de enseñanza y en cómputo. Otro formato de evaluación para programas educativos lo presentan Bernardino de Campos y Elliot⁵⁶. Se manejan dos categorías: los aspectos pedagógicos instruccionales y los aspectos técnico operacionales. Cada categoría presenta subcategorías, como identificación del programa, contenido, metodología e interactividad (primera categoría); documentación, interfaz y recursos computacionales (segunda categoría).

VALIDACIÓN DE SOFTWARE EDUCATIVO POR EXPERTOS.

⁵⁴ Galvis, P.A. Ingeniería de software educativo. 1992.

⁵⁵ Op. cit.

⁵⁶ Bernardino de Campos, G.H. y Elliot G. Evaluación de productos educativos por computadora en "Tecnología y Comunicaciones Educativas". Año 6, num.18. 1991.

A continuación se presentan los lineamientos recomendables para la evaluación de prototipos. Esta propuesta surge del análisis de cada una de las etapas del proceso de desarrollo de un uso educativo de la computadora: diseño instruccional, usos y recursos de la computadora, interfaz al usuario y diseño gráfico. Para el tutorial cella se construyeron los instrumentos de evaluación pertinentes, así como la aplicación de los procedimientos y análisis de datos adecuados.

EXPERTO EN DISEÑO INSTRUCCIONAL	
El análisis de necesidades justifica el desarrollo de un nuevo programa educativo	<input type="radio"/>
La meta instruccional es clara y alcanzable	<input type="radio"/>
El sistema de producción refleja los insumos necesarios al programa educativo	<input type="radio"/>
Se consideran las características de la población meta	<input type="radio"/>
El planteamiento de los objetivos es producto del análisis instruccional	<input type="radio"/>
La secuencia instruccional es producto del análisis de tareas	<input type="radio"/>
Las estrategias de aprendizaje corresponden al tipo de aprendizaje que se está propiciando	<input type="radio"/>
Los medios instruccionales que se plantean están acordes al contenido	<input type="radio"/>
Los criterios de evaluación son coherentes al tipo de conocimiento manejado	<input type="radio"/>

EXPERTO EN DISEÑO DE INTERFAZ AL USUARIO	
Se encuentran consideradas las características de los usuarios, en cuanto al nivel de conocimientos previos y contexto social.	<input type="radio"/>
La interfaz centra al usuario en la tarea de aprendizaje correspondiente	<input type="radio"/>
El tipo de interacción que realiza el usuario es la conveniente a la tarea de aprendizaje presentada	<input type="radio"/>
La selección de la plataforma de cómputo para el desarrollo del producto es la correcta	<input type="radio"/>
Los mecanismos de interacción con que opera el usuario son los adecuados al objetivo del programa	<input type="radio"/>

EXPERTO EN CÓMPUTO	
DISPOSITIVOS	
Se encuentran identifica las características de los usuarios	<input type="radio"/>
La forma de acceso al programa es la más adecuada al contexto de uso	<input type="radio"/>
La decisión sobre el equipo de cómputo tanto de desarrollo como de uso final, es la óptima.	<input type="radio"/>
La selección de la plataforma de cómputo para el desarrollo del producto es la correcta.	<input type="radio"/>

La determinación sobre la colocación de las computadoras en la biblioteca es acorde al objetivo de uso.	
La administración de la forma de uso es amplia y completa.	<input type="radio"/>
Recursos	
La selección de la herramienta de autoría utilizada en el desarrollo es la apropiada	<input type="radio"/>
La digitalización de los recursos sonoros ha sido de calidad y es acorde con el objetivo educativo	<input type="radio"/>
La digitalización de imágenes, tanto fijas como en movimiento, han sido de calidad y acordes con el objetivo educativo.	<input type="radio"/>

EXPERTO EN DISEÑO GRÁFICO	
La diagramación de las zonas interactivas y constantes de diseño es la apropiada al programa educativo	<input type="radio"/>
La distribución de los gráficos (fotografía, ornamentos, ilustración) es la adecuada al programa educativo	<input type="radio"/>
El color empleado ayuda visualmente a comprender los mensajes	<input type="radio"/>
La colocación del texto facilita la comprensión del contenido	<input type="radio"/>
La cantidad de texto manejado es la óptima	<input type="radio"/>
La tipografía utilizada es la más recomendable	<input type="radio"/>

La evaluación de los prototipos por parte de los distintos especialistas que de manera interdisciplinaria colaboraron en su desarrollo, es la parte esencial de cualquier software educativo. Recuérdese que en el tutorial CELLA la colaboración de los especialistas no fue el trabajo directo, sino asesorías en distintas etapas del desarrollo. Cabe notar que existen algunos elementos que se evalúan desde distintas ópticas. Las características de los usuarios potenciales y la tarea de aprendizaje, son factores que debe tener presente el experto en diseño instruccional, en cómputo y en diseño de interfaz. No se debe perder de vista al usuario final ni el objetivo de aprendizaje que se pretende cubrir. Y sobre todo no se debe olvidar que lo primero es lo educativo y lo segundo lo computacional.

Asimismo, la selección del equipo de cómputo, junto con la plataforma y la herramienta de autoría a emplear, no únicamente lo supervisa el especialista en cómputo, sino también el encargado de validar los dispositivos generales de uso que tendrá el programa; aunque en éste caso en particular tuvo que adecuarse a los equipos de la SEP.

De esta manera se presentaron los indicadores que todo productor de un software educativo debe efectuar para evaluar el mismo.

CAPÍTULO IV

PROYECCIÓN A FUTURO

4.1 FUTURO DE LA EAC.

La EAC ha sido probada como una herramienta efectiva en el proceso de enseñanza - aprendizaje, y la implementación de EAC en un ambiente educativo de escuelas publicas, universidades, negocios e incluso en el hogar no tiene precedente de ningún otro fenómeno educativo. Sin embargo, el aprendizaje computarizado parece destinado a jugar un constante y largo rol en la educación en el futuro. Las computadoras están continuamente introduciendose en el ambiente educativo en un numero cada vez mayor y el poder del hardware continúa creciendo. En tanto los productores de software aprendan a hacer software más efectivo y fácil de usar en habilidades básicas de enseñanza; así como en el desarrollo de pensamiento más complejo, la computadora continuará jugando un rol cada vez más importante en la educación.

EN EL NIVEL MEDIO BÁSICO.

En los primeros años del próximo siglo continuará existiendo, para las escuelas secundarias, gran parte de la infraestructura actual. Los alumnos aún trabajarán primero con el profesor de una materia y luego con el de otra a lo largo del día de clase. Es muy poco probable que para la primer década se hagan cambios fundamentales en el sistema, por lo menos en lo que respecta al país. Hawkrigde considera que en Inglaterra y Estados Unidos, probablemente el salón de clases

común tendrá de 10 a 20 PC's especialmente adaptadas como procesadores de palabras. Entre las aplicaciones que Hawkrige considerara valiosas habrá un diccionario automatizado, que despliega en la pantalla la ortografía y, si el alumno lo solicita, el significado y la etimología de cualquier palabra pronunciada ante el micrófono. Los estudiantes se encontrarán también con que el profesor puede regular las computadoras para "gradúen" su tarea, es decir, para que la analicen en un nivel de diagnóstico y le asignen una calificación provisional. Los estudiantes podrán apelar contra ésta calificación, pues sabrán que las computadoras califican de acuerdo a un conjunto limitado de criterios. Se beneficiarán también con la manera nítida en que la máquina indica sus errores y la forma de corregirlos. El profesor de determinada materia contará con una colección de software educativo. Será probable que en algunas materias el alumno aprenda sólo mediante la computadora. Los alumnos podrán solicitar programas adicionales y llevar software a casa para finalizar sus tareas. Espera también que el docente de computación disponga de un aula muy bien equipada. Los niños aprenderán elementos de programación, pero no demasiados, pues en los negocios y la industria la mayor parte de la programación será asistida por computadora. Así Hawkrige expresa su idea de cómo se utilizará la tecnología en el próximo siglo en cuanto a las escuelas secundarias. Cada aula continuará teniendo pizarrón, los libros aún serán importantes y los alumnos seguirán utilizando papel y lápiz. Los laboratorios contendrán todavía un gran número de dispositivos, pero las computadoras resultarán valiosas especialmente en la enseñanza de ciencias. En cuanto a los alumnos prevé que no se aficionarán a la nueva tecnología tanto como los que vengan algunos años después, pero le parece muy probable que a principios del próximo siglo los estudiantes de secundaria aprenderán con rapidez y facilidad

empleando maquinas con interfaces "amigables"; aunque persiste la duda de que lo puedan hacer de igual manera con sus profesores.

POSIBILIDADES PARA EL FUTURO.

Algunas autoridades dentro de la materia, tienen predicho que los avances tecnológicos en hardware y software resultarán en una generación de computadoras capaz de simular el pensamiento humano. Los sistemas de computo tal vez puedan algún día ser capaces de inferir, razonar, aprender y comprender un diálogo humano continuo (Galdwin, 1984).

El imaginar un estudiante con una computadora de bolsillo tal vez del tamaño de una tarjeta de crédito. En la escuela, el hogar o la oficina la tarjeta puede ser insertada en una terminal que pueda acceder un amplio conjunto de datos que provengan de sitios remotos en todo el mundo. Intuitivamente o basado en las necesidades del momento los estudiantes pueden explorar grandes centros de información, la cual puede ser presentada a todo color y en movimiento, impresa o copiada a un archivo personal para usarse posteriormente. Los estudiantes podrían ser capaces de conversar dentro de un sistema con el lenguaje natural, no solo expresando sus necesidades sino también permitiendo a la computadora hacer sugerencias y diagnosticar necesidades y áreas de posible interés

4.2 PROYECCIÓN A FUTURO.

La idea de llevar computadoras a las escuelas sin otros elementos, como el entrenamiento de maestros y los programas de cómputo, ha sido rebasada en gran medida. Actualmente se discute la manera en que las nuevas tecnologías de la información y la comunicación pueden tener un impacto real en las diferentes actividades que se realizan en las instituciones educativas, como promover el uso y manejo de información, permitir situaciones didácticas más enriquecedoras, favorecer la participación activa y la actualización del alumno, facilitar la actualización y superación profesional del profesor, apoyar la experimentación e investigación y, en suma mejorar y modernizar la educación.

Para acercarse a resolver estos requerimientos se necesitan docentes con una visión general de la informática educativa, que estén motivados a explorar con ella y que además cuenten con las habilidades mínimas que les permitan operar con facilidad las herramientas computacionales.

Por otro lado, para que estén en posibilidad de poner en práctica dichos conocimientos es necesario que tengan acceso a espacios donde, a partir de sus experiencias, formulen propuestas fundamentales susceptibles de generalizarse, y participen en la producción de apoyos didácticos como el software educativo.

El modelo adecuado para crear estos escenarios es el Laboratorio. Para 1990 el CPAR había creado 82 Laboratorios de Informática Educativa en las Escuelas Normales de nuestro país. Es importante y significativo el proceso de llegar a una solución integral y la participación de docentes y directivos en la producción y desarrollo del proyecto académico que sustenta las actividades de y entre laboratorios.

Como se mencionó en la introducción y en el capítulo anterior, esta versión CELLA 1.1 será la base para desarrollar nuevas versiones aumentadas y / o actualizadas, esto puede ir desde ampliar la base que contiene las preguntas para las evaluaciones, personalizar las evaluaciones y si se requiere también el contenido, desarrollar nuevas unidades o incluso un glosario hipertexto con los términos más comunes, etc., hasta incluir multimedia cuando los equipos lo permitan. Estas modificaciones podrán realizarlas los profesores dentro del programa COEEBA⁵⁷. Se planea también que esta versión sirva de base a otras desarrolladas para niveles superiores, específicamente existe un proyecto de realizar una versión para el segundo año de preparatoria.

Uno de los objetivos principales de esta tesis es que el desarrollo de este producto sirva como referencia a otros productores y desarrolladores de software educativo, esto es en especial importante ya que la EAC es relativamente nueva en el país, y por ello lamentablemente no existe la literatura suficiente que documente experiencias, metodologías de desarrollo, metodologías de producción, etc. Así la producción del tutorial CELLA incrementa las pocas experiencias de desarrollo y producción de software educativo del país.

⁵⁷ Se considera que ellos son los adecuados ya que dentro del programa COEEBA existe un curso de programación en lenguaje Pascal y también bases para desarrollo de PCE. Aunque cualquier profesor que tenga experiencia en el área podrá modificarlo de acuerdo a sus necesidades.

CONCLUSIONES

Entre las ventajas más importantes de la EAC, se encuentran: la capacidad de mantener la atención del sujeto activa durante el proceso de aprendizaje, así como que cada estudiante tiene la posibilidad de aprender a su propio ritmo. La dificultad se presenta en tanto no es fácil medir su efectividad.

La efectividad ha sido definida de varias maneras. Puede ser medida por el rigor de los conceptos aprendidos y también por la cantidad de materia aprendida durante un determinado periodo de tiempo. Pero también tiene que ver la efectividad con el grado de retención de lo aprendido. Además hay que considerar como elemento de efectividad, entre otros, el efecto que puede producirse en el cambio de actitud del alumno hacia la computadora, si es considerada como medio educativo o meramente cultural.

Muy pocos proyectos de EAC han sido evaluados de una manera sistemática y controlada, como lo ha sido TICCIT y PLATO. Pero en general se desprenden las siguientes conclusiones con respecto al uso de la EAC:

1. Mejora el aprendizaje; y si esto no es logrado no presenta diferencia con respecto al aprendizaje en la clase tradicional.
2. Reduce el tiempo de aprendizaje comparado con la clase regular
3. Produce en la actitud del estudiante un efecto favorable al uso de la computadora en la enseñanza.

Puede decirse, que el grado de la efectividad de la EAC es, en general, altamente positivo. Sin embargo, no hay que ocultar también ciertos factores negativos al valorar sus posibilidades.

A pesar de sus ventajas, la EAC encuentra ciertas dificultades para su implantación, que se deben fundamentalmente a tres factores:

1. el costo
2. la situación primitiva en que se encuentra la EAC, (cuando menos en el país) y
3. la actitud de profesores y responsables de la educación.

El sector educativo nacional se encuentra frente a una nueva tecnología con un gran potencial para la educación. Es verdad que aún no se sabe lo suficiente sobre la forma de aprovecharla plenamente o sobre como evitar sus "peligros", pero se está acumulando experiencia. Se observa el surgimiento y afianzamiento de un "sistema multimedia" a medida que se incorporan "medios" que se combinan de una manera poderosa, y que ofrecen la capacidad de interactuar con los estudiantes a cualquier nivel. El punto es preguntarse sí la educación aprovechara ésta circunstancia.

No es conveniente dejarse llevar por la moda de la "utopía computacional", pero tampoco por el pesimismo. Lo que se necesita es decisión, optimismo moderado, dedicación y, por supuesto, imaginación y creatividad, para poder llevar a cabo este gran proyecto y que pueda ofrecer un provecho real e integral a los alumnos.

Aún así los productores de EAC deberán de continuar informándose sobre nuevas propuestas de diseño, desarrollo e implementación de EAC y revisar las investigaciones recientes sobre las características de EAC efectiva. Otra buena

forma de aprender es “usar” y evaluar tantos nuevos programas de software educativo como sea posible y visitar locaciones que hagan efectivo el uso de EAC. Las “visitas” se aplican, por supuesto, también a Internet, ya que a través de sus cada vez más numerosas páginas WEB ofrece software educativo en sus diferentes tipos; debe considerarse que el hecho de que estos productos estén “publicados” no significa que sean buenos productos, por lo tanto deben “usarse” y evaluarse para incrementar la experiencia en todos los puntos de desarrollo (diseño, interfaz, técnicas de enseñanza, evaluación, etc.).

La conclusión de esta tesis es que la única manera de aplicar de manera efectiva la EAC en México es, producir software educativo, ya que sólo produciendo se puede acumular la experiencia tan necesaria para encontrar y mejorar las técnicas que permitan desarrollar productos adecuados al país, esto dará como resultado productos de calidad cada vez mayor y al obtenerlos se cumplirá el objetivo más importante de la EAC que es precisamente ENSEÑAR.

BIBLIOGRAFÍA

Hudson, Keith. Enseñanza asistida por ordenador.
España, Ed. Diaz de Santos, 1986.

Hawkrige D., Informática y educación.
Argentina, Ed. Kapelusz, 1983.

Pfeiffer A., Informática y escuela.
España, Ed. FUNDESCO, 1985.

Price, Robert V., Computer-Aided Instruction.
U.S.A., Ed. Brooks/Cole Publishing Co, 1991.

Price, Robert V., An historical perspective on the design of computer - assisted instruction: lessons from the past
computer-Aided Instruction.
U.S.A., 1989.

Bauersfeld, Penny. Software by Design.
New York, Ed. M&T Books, 1994.

Van der Mollen, Tina. The Hyper Ed Handbook.
Cupertino, Apple Computer/Intellimation, 1990.

Schneiderman, Ben. Design in the user interface.
Addison - Wesley, 1992.

Bernardino de Campos, G.H: y G. Elliot. Evaluación de productos educativos para computadora
en: Tecnología y Comunicaciones Educativas. Año 6, num. 18, 1991.

Galvis, P.A. Ingeniería de software educativo.
Bogotá, Ed. Uniandes, 1992.

Grogono P. Programación en Pascal.
México, Ed. Addison-Wesley, 1986.

SEP, Plan y programas de estudio 1993
Educación Básica Secundaria . México, 1993.

Ville, Claude A. Biología.

México, Ed. Interamericana, 1986.

Alonso Tejeda E. La ciencia de la vida.

México, Ed. McGraw-Hill, 1985.

Castellanos Iturriaga J. Biología.

México, Ed. Esfinge, 1980.

Gros, B. , Aprender mediante el ordenador

1989.

Markle, God frames and bad

New York, 1964.

ILCE, COEEBA. La microcomputadora como auxiliar didáctico en el aula ..
en Computación Electrónica en la Educación Básica. México, ILCE. 1991.

CISE. Usos Educativos de la Computadora

México, UNAM, 1994.

Dick, W. y Carey, L. Diseño sistemático de la instrucción.

Bogotá, Voluntad Editores, 1978.

Woolfolk, A., Educational Pshychology

Englewood Cliffs, News, Ed. Prentice - Hall, Inc., 1987.

Brown, W. et al, Instrucción audiovisual

México, Ed. Trillas, 1981.

Schyfter, L. G., Aspectos generales de la evaluación del rendimiento escolar
en: F. García. Paquete de autoenseñanza de evaluación del aprovechamiento escolar.
México, CISE-UNAM, 1983.

Norman, Donald. The Design of Everyday Things

1990.

Apple Computer, Inc. Human Interface Guidelines: The Apple Desktop Interface.

1987.

Hewlett Packard. Liso del color.
1993.

Bañuelos M. Ana Ma. Diseño Instruccional
México, CISE - UNAM, 1994.

Eco, H. Como se hace una tesis
Barcelona, Ed. Gedisa, 1992.

IBM, Story Board Level User Guide
Florida, IBM Corp., 1990.

IBM, Story Board Level 2.0 Getting Started/ Tutorial Guide
Florida, IBM Corp., 1992.

Forester, T. La sociedad de la alta tecnología
México, Ed. sXXI, 1992.

Hudson K. Introducing CAL: a practical guide to Writing Computer - Assisted Learning Programs.
London, Chapman and Hall, 1984.

Memorias de la séptima conferencia internacional
"Las computadoras en instituciones de educación y de investigación"
Pantallas, formatos y sugerencias para empezar en sistemas tutoriales
México, UNAM, 1991.

Suppes, P. & Morningstar, M. CAL at Stanford, 1966-68: Data, models and evaluation of arithmetic programs.
New York, Academic Press, 1972.

Vaquero, A. La informática aplicada a la enseñanza
Madrid, Edudema S.A., 1987.

Poder Ejecutivo Federal
Plan Nacional de Desarrollo 1983 - 1988

Poder Ejecutivo Federal
Programa Nacional de educación, cultura y deporte 1984 - 1988

Poder Ejecutivo Federal
Programa para la modernización educativa 1984 - 1988

APENDICES

A. PROGRAMAS

A.1 PROGRAMA CELULA.PAS

A.2 LECCIÓN 2 DEL TUTORIAL CELLA

B. PANTALLAS DEL TUTORIAL CELLA

C. MANUAL DE INSTALACIÓN

```

program entrada(input,output);
{$M 28672,0,28672}
uses
  crt,dos,printer;
const
  maxalum = 30;
type
  student=record
    nombre : string[30];
    edad : string[2];
    leccion : integer;
    last_fec: string[12];
    clave : string[3];
    calif : integer;
  end;
  resultado = record
    suma : integer;
  end;
var
  archalum : file of student;
  puntos : file of resultado;
  no : integer;
  alumno : student;
  punt_t : resultado;
  modoTi :integer;
  t :string[40];
  m : char;

procedure fecha;
const
  dias:array [0..6] of string[9]=('DOMINGO','LUNES','MARTES','MIERCOLES',
    'JUEVES','VIERNES','SABADO');
var
  d,m,dow,a:word;
  di,me:string[2];
  an:string[4];
  NUM : INTEGER;

begin
  getdate(a,m,d,dow);
  textbackground(1);
  textcolor(15);
  gotoxy(2,16);
  writeln('EMPIEZA TU CLASE DEL ',dias[dow], ' ',d,'/',m,'/',a);
  str(d:2,di);
  str(m:2,me);
  str(a:4,an);
  alumno.last_fec:=concat(di,'/',me,'/',an);
end;

procedure accesol;
var
  d,m,a,dow:word;
  i,e,g :integer;
  tem,clavel: string[3];
  di,me,an:string[2];
  seguro,sale: char;
  l:real;
begin
  clrscr;

```

```

reset (archalum);
if filesize(archalum)=0 then
  exit
else
  begin
    repeat
      gotoxy(9,6);
      writeln('PULSA TU CLAVE POR FAVOR => ');
      textbackground(0);
      gotoxy(18,8);
      write('  ');
      gotoxy(18,8);
      textcolor(0);
      readln(clavel);
      e:=0;
      sale:='j';
      i:=filesize(archalum);
      while not eof(archalum) do
        begin
          seek(archalum,e);
          read(archalum,alumno);
          if alumno.clave=clavel then
            begin
              sale:='T';

              end;
              e:=e+1;
            end;
          until sale<>'j';
          e:=0;
          seek(archalum,i);
          i:=filepos(archalum);
          with alumno do
            begin
              getdate(a,m,d,dow);
              str(d:2,di);
              str(m:2,me);
              str(a:4,an);
              if last_fec = concat(di,'/',me,'/',an) then
                begin
                  repeat
                    textbackground(1);
                    textcolor(15);
                    gotoxy(4,20);
                    sale:='n';
                    write('ACCESO DENEGADO POR EL DIA DE HOY');
                    until keypressed;
                  end
                else
                  begin
                    textbackground(1);
                    textcolor(15);
                    gotoxy(2,20);
                    writeln('EMPIEZA TU CLASE DEL ',d,'/',m,'/',a);
                    gotoxy(13,18);
                    sale:='s';
                    leccion:=leccion+1;
                    i:=filepos(archalum);
                    last_fec := concat(di,'/',me,'/',an);

```

```

modori:=lastmode;
textmode(1);
textcolor(15);
repeat
  textbackground(1);
  clrscr;
  gotoxy(6,7);
  writeln('■■■ B I O L O G I A   2 ■■■');
  gotoxy(15,13);
  writeln('C E L L A');
  gotoxy(4,24);
  writeln('PRESIONA <E N T E R> PARA CONTINUAR');
  sale:=readkey;
until sale=chr(13);
textcolor(15);
textbackground(1);
clrscr;
gotoxy(4,3);
write('ES LA PRIMERA VEZ QUE ESTAS EN');
gotoxy(12,20);
write('C E L L A (S/N) => ');
readln(seguro);
if upcase(seguro)='S'
then
begin
  e:=0;
  clrscr;
  reset(archalum);
  repeat
    gotoxy(4,4);
    writeln('ESCOGE UNA CLAVE CON TRES LETRAS => ');
    gotoxy(20,6);
    textbackground(0);
    write(' ');
    textbackground(1);
    textcolor(11);
    gotoxy(20,6);
    readln(clavel);
    if filesize(archalum)<>0 then
      begin
        sale:='a';
        while not eof(archalum) do
          begin
            seek(archalum,e);
            read(archalum,alumno);
            if clavel=alumno.clave then
              sale:='h';
            e:=e+1;
          end;
        end
      end
    else
      sale:='z';
  until sale<>'h';
  with alumno do
    begin
      textbackground(1);
      textcolor(15);
      gotoxy(13,9);
      writeln('COMO TE LLAMAS ? => ');
      gotoxy(4,11);

```

```

    textbackground(0);
    write('
    gotoxy(4,11);
    textbackground(1);
    textcolor(11);
    readln(nombre);
    gotoxy(8,14);
    textbackground(1);
    textcolor(15);
    write('CUANTOS AÑOS TIENES ? => ');
    textbackground(0);
    write(' ');
    textbackground(1);
    textcolor(11);
    gotoxy(33,14);
    readln(edad);
    fecha;
    clave:=clave1;
    leccion:=1;
    write(archalum,alumno);
    repeat
        gotoxy(15,18);
        write('BIENVENIDO');
        i:=length(nombre);
        l:=((40-i)/2);
        i:=trunc(l);
        textcolor(11);
        gotoxy(i,20);
        write(nombre);
        textcolor(15);
        gotoxy(15,22);
        write('COMENZAMOS');
    until keypressed;

    end;(*endwith*)
(*close(archalum);*)
swapvectors;
exec('c:\cella\st.exe','c:\cella\L1GOOD');
swapvectors;
textmode(modori);
swapvectors;
exec('c:\cella\eval1.exe','');
swapvectors;
reset(puntos);
seek(puntos,0);
read(puntos,punt_t);
g:=punt_t.suma;
close(puntos);
if g>=8 then
    alumno.calif:=g
else
    alumno.leccion:=0;
close(archalum);
modori:=lastmode;
textmode(1);
end(*endif*)
else
begin
    if upcase(seguro)='N' then
begin

```

```
if sale='s' then
begin
  case leccion of
    1:begin
      swapvectors;
      exec('c:\cella\st.exe','c:\cella\L1GOOD');
      swapvectors;
      textmode(modori);
      swapvectors;
      exec('c:\cella\eval1.exe','');
      swapvectors;
      reset(puntos);
      seek(puntos,0);
      read(puntos,punt_t);
      g:=punt_t.suma;
      close(puntos);
      if g>=8 then
        begin
          alumno.leccion:=1;
          alumno.calif:=g;
        end
      else
        alumno.leccion:=0;
      end;
    2:begin
      swapvectors;
      exec('c:\cella\st.exe','c:\cella\L2GOOD');
      swapvectors;
      textmode(modori);
      swapvectors;
      exec('c:\cella\eval2.exe','');
      swapvectors;
      reset(puntos);
      seek(puntos,0);
      read(puntos,punt_t);
      g:=punt_t.suma;
      close(puntos);
      if g>=8 then
        begin
          alumno.leccion:=2;
          alumno.calif:=alumno.calif+g;
        end
      else
        alumno.leccion:=1;
      end;
    3:begin
      swapvectors;
      exec('c:\cella\st.exe','c:\cella\L3GOOD');
      swapvectors;
      textmode(modori);
      swapvectors;
      exec('c:\cella\eval3.exe','');
      swapvectors;
      reset(puntos);
      seek(puntos,0);
      read(puntos,punt_t);
      g:=punt_t.suma;
      close(puntos);
```

```

if g>=8 then
    Apndice A.1
    begin
        alumno.leccion:=3;
        alumno.calif:=alumno.calif+g;
        clrscr;
        t:='» C . I . A. «';
        i:=length(t);
        l:=(40-i)/2;
        i:=trunc(l);
        gotoxy(i,2);
        writeln(t);
        t:='Otorga el presente ';
        i:=length(t);
        l:=(40-i)/2;
        i:=trunc(l);
        gotoxy(i,4);
        writeln(t);
        t:='D I P L O M A';
        i:=length(t);
        l:=(40-i)/2;
        i:=trunc(l);
        gotoxy(i,8);
        writeln(t);
        t:=alumno.nombre;
        i:=length(t);
        l:=(40-i)/2;
        i:=trunc(l);
        gotoxy(i-3,10);
        write('A ',t);
        t:='como reconocimiento por haber terminado ';
        i:=length(t);
        l:=(40-i)/2;
        i:=trunc(l);
        gotoxy(i,12);
        write(t);
        t:='satisfactoriamente el tutorial CELLA ';
        i:=length(t);
        l:=(40-i)/2;
        i:=trunc(l);
        gotoxy(i,14);
        write(t);
        repeat
            until keypressed;
        end
        else
            alumno.leccion:=2;
        end;
    end;
end;
end;
write(archalum,alumno);
end;(*with*)
close(archalum);
end;
end;

end;(*endif*)
textmode(modori);
end;(*acceso*)

```

```
begin
  assign(archalum, 'alumnos.dat');
  assign(puntos, 'temp.dat');
  accesol;
  CLRSCR;
  textmode(modori);
  write('■■■■ FIN CELLA ■■■■ ');
end.
```

Apendice A.1

NO.	LABEL	COMMAND	Appendice A.2		TIME	WAIT	METHOD	SP
			PARAMETERS					
1	100	/DISPLAY*	II	.PIC-	SLOW	0	FADE	
2	101	/DISPLAY*	CONTI	.PIC-	SLOW	0	FADE	
3	67	/DISPLAY*	CAR4	.PIC-	SLOW	0	FADE	
4	MAIN	/DISPLAY*	KLM2	.PIC-	FAST	0	FADE	
5		/BUTTON*A	437-111	597-139	FAST	0		
6		/BUTTON*B	438-150	599-163	FAST	0		
7		/BUTTON*C	436-174	600-195	FAST	0		
8		/BUTTON*D	437-208	597-223	FAST	0		
9		/BUTTON*E	435-233	599-249	FAST	0		
10		/BUTTON*F	436-257	600-270	0	0		
11		/BUTTON*G	436-281	598-298	0	0		
12		/BUTTON*H	434-305	601-320	0	0		
13		/BUTTON*I	436-329	600-343	0	0		
14		/BUTTON*J	433-352	600-369	0	0		
15		/BUTTON*S	436-381	597-398	0	0		
16		/INPUT*	1		0	0		
17		/IF*A	GOTO PARTA		0	0		
18		/IF*B	GOTO PARTB		0	0		
19		/IF*C	GOTO PARTC		0	0		
20		/IF*D	GOTO PARTD		0	0		
21		/IF*E	GOTO PARTE		0	0		
22		/IF*F	GOTO PARTF		0	0		
23		/IF*G	GOTO PARTG		0	0		
24		/IF*H	GOTO PARTH		0	0		
25		/IF*I	GOTO PARTI		0	0		
26		/IF*J	GOTO PARTJ		0	0		
27		/IF*S	GOTO 87		0	0		
28	PARTA	/*MEMBRANA PLASMATICA			0	0		
29		/DISPLAY*	P13	.PIC-	0	0	FADE	
30		/GOSUB*	00		0	0		
31		/IF*1	GOTO MAIN		0	0		
32		/IF*2	GOTO P2		0	0		
33	P2	/DISPLAY*	P13B	.PIC-	0	0	FADE	
34		/GOSUB*	00		0	0		
35		/IF*1	GOTO PARTA		0	0		
36		/IF*2	GOTO P3		0	0		
37	P3	/DISPLAY*	P13C	.PIC-	0	0	FADE	
38		/GOSUB*	00		0	0		
39		/IF*1	GOTO P2		0	0		
40		/IF*2	GOTO MAIN		0	0		
41		/RETURN*			0	0		
42	PARTB	/*MITOCONDRIA			0	0		
43		/DISPLAY*	PRUE11B	.PIC-	0	0	FADE	
44		/GOSUB*	00		0	0		
45		/IF*1	GOTO MAIN		0	0		
46		/IF*2	GOTO M1		0	0		
47	M1	/DISPLAY*	PRUE11C	.PIC-	0	0	FADE	
48		/GOSUB*	00		0	0		
49		/IF*1	GOTO PARTB		0	0		
50		/IF*2	GOTO MAIN		0	0		

		Apndice A.2					SP
NO.	LABEL	COMMAND	PARAMETERS	TIME	WAIT	METHOD	
51		/RETURN*		0	0		
52	PARTC	/*RETICULO ENDOPLASMAT		0	0		
53		/DISPLAY*	P17 .PIC-	0	0	FADE	
54		/GOSUB*	00	0	0		
55		/IF*1	GOTO MAIN	0	0		
56		/IF*2	GOTO R1	0	0		
57	R1	/DISPLAY*	RIB1 .PIC-	0	0	FADE	
58		/GOSUB*	00	0	0		
59		/IF*1	GOTO PARTC	0	0		
60		/IF*2	GOTO R2	0	0		
61	R2	/DISPLAY*	RIB1A .PIC-	0	0	FADE	
62		/GOSUB*	00	0	0		
63		/IF*1	GOTO R2	0	0		
64		/IF*2	GOTO MAIN	0	0		
65		/RETURN*		0	0		
66	PARTD	/*APARATO DE GOLGI		0	0		
67	G1	/DISPLAY*	F1-7 .PIC-	0	0	FADE	
68		/GOSUB*	00	0	0		
69		/IF*1	GOTO PARTD	0	0		
70		/IF*2	GOTO MAIN	0	0		
71		/RETURN*		0	0		
72	PARTE	/*NUCLEOLO		0	0		
73		/DISPLAY*	P8 .PIC-	0	0	FADE	
74		/GOSUB*	00	0	0		
75		/IF*1	GOTO MAIN	0	0		
76		/IF*2	GOTO MAIN	0	0		
77		/RETURN*		0	0		
78	PARTF	/*CROMOSOMA		0	0		
79		/DISPLAY*	PRUE12 .PIC-	0	0	FADE	
80		/GOSUB*	00	0	0		
81		/IF*1	GOTO MAIN	0	0		
82		/IF*2	GOTO CR1	0	0		
83	CR1	/DISPLAY*	P12 .PIC-	0	0	FADE	
84		/GOSUB*	00	0	0		
85		/IF*1	GOTO PARTF	0	0		
86		/IF*2	GOTO CR2	0	0		
87	CR2	/DISPLAY*	PRUE12 .PIC-	0	0	FADE	
88		/GOSUB*	00	0	0		
89		/IF*1	GOTO CR1	0	0		
90		/IF*2	GOTO MAIN	0	0		
91		/RETURN*		0	0		
92	PARTG	/*NUCLEO		0	0		
93		/DISPLAY*	PRUE8 .PIC-	0	0	FADE	
94		/GOSUB*	00	0	0		
95		/IF*1	GOTO MAIN	0	0		
96		/IF*2	GOTO MAIN	0	0		
97		/RETURN*		0	0		
98	PARTH	/*VACUOLA		0	0		
99		/DISPLAY*	PRUE19 .PIC-	0	0	FADE	
100		/GOSUB*	00	0	0		

		Appendix A 2					SP
NO.	LABEL	COMMAND	PARAMETERS	TIME	WAIT	METHOD	
101		/IF*1	GOTO MAIN	0	0		
102	AS	/IF*2	GOTO V1	0	0		
103		/GOSUB*	00	0	0		
104		/IF*1	GOTO AS	0	0		
105		/IF*2	GOTO V1	0	0		
106	V1	/DISPLAY*	PRUE19A .PIC-	0	0	FADE	
107		/GOSUB*	00	0	0		
108		/IF*1	GOTO AS	0	0		
109		/IF*2	GOTO MAIN	0	0		
110		/RETURN*		0	0		
111	PARTI	/*LISOSOMA		0	0		
112		/DISPLAY*	PRUE20 .PIC-	0	0	FADE	
113		/GOSUB*	00	0	0		
114		/IF*1	GOTO MAIN	0	0		
115		/IF*2	GOTO MAIN	0	0		
116		/RETURN*		0	0		
117	PARTJ	/*RIBOSOMA		0	0		
118		/DISPLAY*	CE4A .PIC-	0	0	FADE	
119		/GOSUB*	00	0	0		
120		/IF*1	GOTO MAIN	0	0		
121		/IF*2	GOTO MAIN	0	0		
122		/RETURN*		0	0		
123	87	/GOSUB*	PM	SLOW	0		
124	88	/DISPLAY*	ANSWA .PIC-	SLOW	0	FADE	
125		/GOSUB*	PREG	0	0		
126		/IF*C	GOTO O1	0	0		
127		/IFNOT*C	GOTO T1	0	0		
128	O1	/DISPLAY*	MHG .PIC-	0	0	INSTAN	
129		/GOTO*	89	0	0		
130	T1	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN	
131		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN	
132		/GOTO*	88	0	0		
133	89	/DISPLAY*	ANSWB .PIC-	SLOW	0	CHECKE	
134		/GOSUB*	PREG	0	0		
135		/IF*B	GOTO O2	0	0		
136		/IFNOT*B	GOTO T2	0	0		
137	O2	/DISPLAY*	MHG .PIC-	0	0	INSTAN	
138		/GOTO*	90	0	0		
139	T2	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN	
140		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN	
141		/GOTO*	89	0	0		
142	90	/DISPLAY*	ANSWC .PIC-	SLOW	0	FADE	
143		/GOSUB*	PREG	0	0		
144		/IF*A	GOTO O3	0	0		
145		/IFNOT*A	GOTO T3	0	0		
146	O3	/DISPLAY*	MHG .PIC-	0	0	INSTAN	
147		/GOTO*	91	0	0		
148	T3	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN	
149		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN	
150		/GOTO*	90	0	0		

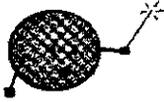
NO.	LABEL	COMMAND	Appendix A.2		TIME	WAIT	METHOD	SP
			PARAMETERS					
151	91	/DISPLAY*	ANSWD	.PIC-	SLOW	0	CRUSH	
152		/GOSUB*	PREG		0	0		
153		/IF*C	GOTO O4		0	0		
154		/IFNOT*C	GOTO T4		0	0		
155	O4	/DISPLAY*	MHG	.PIC-	0	0	INSTAN	
156		/GOTO*	92		0	0		
157	T4	/DISPLAY*	TONTO1	.PIC-	0	0	INSTAN	
158		/DISPLAY*	TONTO2	.PIC-	0	0	INSTAN	
159		/GOTO*	91		0	0		
160	92	/DISPLAY*	ANSWE	.PIC-	SLOW	0	CRUSH	
161		/GOSUB*	PREG		0	0		
162		/IF*C	GOTO O5		0	0		
163		/IFNOT*C	GOTO T5		0	0		
164	O5	/DISPLAY*	MHG	.PIC-	0	0	INSTAN	
165		/GOTO*	93		0	0		
166	T5	/DISPLAY*	TONTO1	.PIC-	0	0	INSTAN	
167		/DISPLAY*	TONTO2	.PIC-	0	0	INSTAN	
168		/GOTO*	92		0	0		
169	93	/DISPLAY*	ANSWF	.PIC-	SLOW	0	CRUSH	
170		/GOSUB*	PREG		0	0		
171		/IF*B	GOTO O6		0	0		
172		/IFNOT*B	GOTO T6		0	0		
173	O6	/DISPLAY*	MHG	.PIC-	0	0	INSTAN	
174		/GOTO*	94		0	0		
175	T6	/DISPLAY*	TONTO1	.PIC-	0	0	INSTAN	
176		/DISPLAY*	TONTO2	.PIC-	0	0	INSTAN	
177		/GOTO*	93		0	0		
178	94	/DISPLAY*	ANSWG	.PIC-	SLOW	0	EXPLOD	
179		/GOSUB*	PREG		0	0		
180		/IF*C	GOTO O7		0	0		
181		/IFNOT*C	GOTO T7		0	0		
182	O7	/DISPLAY*	MHG	.PIC-	0	0	INSTAN	
183		/GOTO*	95		0	0		
184	T7	/DISPLAY*	TONTO1	.PIC-	0	0	INSTAN	
185		/DISPLAY*	TONTO2	.PIC-	0	0	INSTAN	
186		/GOTO*	94		0	0		
187	95	/DISPLAY*	ANSWH	.PIC-	SLOW	0	FADE	
188		/GOSUB*	PREG		0	0		
189		/IF*A	GOTO O8		0	0		
190		/IFNOT*A	GOTO T8		0	0		
191	O8	/DISPLAY*	MHG	.PIC-	0	0	INSTAN	
192		/GOTO*	96		0	0		
193	T8	/DISPLAY*	TONTO1	.PIC-	0	0	INSTAN	
194		/DISPLAY*	TONTO2	.PIC-	0	0	INSTAN	
195		/GOTO*	95		0	0		
196	96	/DISPLAY*	ANSWI	.PIC-	SLOW	0	PUSH	
197		/GOSUB*	PREG		0	0		
198		/IF*B	GOTO O9		0	0		
199		/IFNOT*B	GOTO T9		0	0		
200	O9	/DISPLAY*	MHG	.PIC-	0	0	INSTAN	

NO.	LABEL	COMMAND	Apendice A.2		TIME	WAIT	METHOD	SP
			PARAMETERS					
201		/GOTO*	97		0	0		
202	T9	/DISPLAY*	TONTO1 .PIC-		0	0	INSTAN	
203		/DISPLAY*	TONTO2 .PIC-		0	0	INSTAN	
204		/GOTO*	96		0	0		
205	97	/DISPLAY*	ANSWJ .PIC-		SLOW	0	PUSH	
206		/GOSUB*	PREG		0	0		
207		/IF*B	GOTO O10		0	0		
208		/IFNOT*B	GOTO T10		0	0		
209	O10	/DISPLAY*	MHG .PIC-		0	0	INSTAN	
210		/GOTO*	98		0	0		
211	T10	/DISPLAY*	TONTO1 .PIC-		0	0	INSTAN	
212		/DISPLAY*	TONTO2 .PIC-		0	0	INSTAN	
213		/GOTO*	97		0	0		
214	98	/DISPLAY*	ANSWK .PIC-		SLOW	0	REPLAC	
215		/GOSUB*	PREG		0	0		
216		/IF*B	GOTO O11		0	0		
217		/IFNOT*B	GOTO T11		0	0		
218	O11	/DISPLAY*	MHG .PIC-		0	0	INSTAN	
219		/GOTO*	102		0	0		
220	T11	/DISPLAY*	TONTO1 .PIC-		0	0	INSTAN	
221		/DISPLAY*	TONTO2 .PIC-		0	0	INSTAN	
222		/GOTO*	98		0	0		
223	102	/DISPLAY*	CAR5 .PIC-		SLOW	0	FADE	
224	103	/DISPLAY*	CA5 .PIC-		SLOW	0	FADE	
225		/GOSUB*	00		0	0		
226		/IF*1	GOTO 103		0	0		
227		/IF*2	GOTO 104		0	0		
228	104	/DISPLAY*	CA50A .PIC-		SLOW	0	EXPLOD	
229		/GOSUB*	00		0	0		
230		/IF*1	GOTO 103		0	0		
231		/IF*2	GOTO 105		0	0		
232	105	/DISPLAY*	CA50B .PIC-		SLOW	0	FADE	
233		/GOSUB*	00		0	0		
234		/IF*1	GOTO 104		0	0		
235		/IF*2	GOTO 106		0	0		
236	106	/DISPLAY*	CA51 .PIC-		FAST	0	CHECKE	
237		/GOSUB*	00		0	0		
238		/IF*1	GOTO 105		0	0		
239		/IF*2	GOTO 107		0	0		
240	107	/DISPLAY*	CA52 .PIC-		SLOW	0	FADE	
241		/GOSUB*	00		0	0		
242		/IF*1	GOTO 106		0	0		
243		/IF*2	GOTO 108		0	0		
244	108	/DISPLAY*	ELOSO .PIC-		0	0	CRUSH	
245		/BUTTON*A	95-345 194-364		0	0		
246		/BUTTON*B	214-399 332-419		0	0		
247		/BUTTON*C	430-324 512-343		0	0		
248		/BUTTON*S	412-395 510-411		0	0		
249		/INPUT*			0	0		
250		/IF*A	GOTO SRI		0	0		

				Appendix A.2				SP
NO.	LABEL	COMMAND	PARAMETERS	TIME	WAIT	METHOD		
251		/IF*B	GOTO SR2	0	0			
252		/IF*C	GOTO SR3	0	0			
253		/IF*S	GOTO 112	0	0			
254	SR2	/DISPLAY*	PARED .PIC-	SLOW	0	CRUSH		
255		/GOSUB*	00	0	0			
256		/IF*1	GOTO 108	0	0			
257		/IF*2	GOTO 109	0	0			
258	109	/DISPLAY*	PARED2 .PIC-	SLOW	0	CRUSH		
259		/GOSUB*	00	0	0			
260		/IF*1	GOTO SR1	0	0			
261		/IF*2	GOTO 108	0	0			
262		/RETURN*		0	0			
263	SR1	/DISPLAY*	CLOR .PIC-	SLOW	0	CRUSH		
264		/GOSUB*	00	0	0			
265		/IF*1	GOTO 108	0	0			
266		/IF*2	GOTO SR2A	0	0			
267	SR2A	/DISPLAY*	CLOR1 .PIC-	SLOW	0	CRUSH		
268		/GOSUB*	00	0	0			
269		/IF*1	GOTO SR2	0	0			
270		/IF*2	GOTO 108	0	0			
271		/RETURN*		0	0			
272	SR3	/DISPLAY*	CEN1 .PIC-	SLOW	0	EXPLOD		
273		/GOSUB*	00	0	0			
274		/IF*1	GOTO 108	0	0			
275		/IF*2	GOTO 108	0	0			
276		/RETURN*		0	0			
277	112	/GOSUB*	PM	SLOW	0			
278	113	/DISPLAY*	ANSW11 .PIC-	SLOW	0	FADE		
279		/GOSUB*	PREG	0	0			
280		/IF*C	GOTO O12	0	0			
281		/IFNOT*C	GOTO T12	0	0			
282	O12	/DISPLAY*	MHG .PIC-	0	0	INSTAN		
283		/GOTO*	114	0	0			
284	T12	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN		
285		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN		
286		/GOTO*	113	0	0			
287	114	/DISPLAY*	ANSW12 .PIC-	SLOW	0	FADE		
288		/GOSUB*	PREG	0	0			
289		/IF*A	GOTO O13	0	0			
290		/IFNOT*A	GOTO T13	0	0			
291	O13	/DISPLAY*	MHG .PIC-	0	0	INSTAN		
292		/GOTO*	115	0	0			
293	T13	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN		
294		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN		
295		/GOTO*	114	0	0			
296	115	/DISPLAY*	ANSW13 .PIC-	SLOW	0	CRUSH		
297		/GOSUB*	PREG	0	0			
298		/IF*A	GOTO O14	0	0			
299		/IFNOT*A	GOTO T14	0	0			
300	O14	/DISPLAY*	MHG .PIC-	0	0	INSTAN		

						Appendix A.2			SP
NO.	LABEL	COMMAND	PARAMETERS	TIME	WAIT	METHOD			
301		/GOTO*	116	0	0				
302	T14	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN			
303		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN			
304		/GOTO*	115	0	0				
305	116	/DISPLAY*	CAR6 .PIC-	SLOW	0	FADE			
306	117	/DISPLAY*	NEURO .PIC-	SLOW	0	DIAGON			
307		/GOSUB*	00	0	0				
308		/IF*1	GOTO 117	0	0				
309		/IF*2	GOTO 118	0	0				
310	118	/DISPLAY*	NEURO2 .PIC-	FAST	0	SPLIT			
311		/GOSUB*	00	0	0				
312		/IF*1	GOTO 117	0	0				
313		/IF*2	GOTO 119	0	0				
314	119	/DISPLAY*	NEURO3 .PIC-	SLOW	0	REPLAC			
315		/GOSUB*	00	0	0				
316		/IF*1	GOTO 118	0	0				
317		/IF*2	GOTO 120	0	0				
318	120	/DISPLAY*	NEURO4 .PIC-	SLOW	0	FADE			
319		/GOSUB*	00	0	0				
320		/IF*1	GOTO 119	0	0				
321		/IF*2	GOTO 121	0	0				
322	121	/DISPLAY*	NEURO5 .PIC-	SLOW	0	FADE			
323		/GOSUB*	00	0	0				
324		/IF*1	GOTO 120	0	0				
325		/IF*2	GOTO 122	0	0				
326	122	/DISPLAY*	NEURO5A .PIC-	SLOW	0	FADE			
327		/GOSUB*	00	0	0				
328		/IF*1	GOTO 121	0	0				
329		/IF*2	GOTO 123	0	0				
330	123	/DISPLAY*	NEURO6 .PIC-	FAST	0	CRUSH			
331		/GOSUB*	00	0	0				
332		/IF*1	GOTO 122	0	0				
333		/IF*2	GOTO 124	0	0				
334	124	/DISPLAY*	NEURO7 .PIC-	SLOW	0	EXPLOD			
335		/GOSUB*	00	0	0				
336		/IF*1	GOTO 123	0	0				
337		/IF*2	GOTO 125	0	0				
338	125	/DISPLAY*	NEURO8 .PIC-	SLOW	0	FADE			
339		/GOSUB*	00	0	0				
340		/IF*1	GOTO 124	0	0				
341		/IF*2	GOTO 126	0	0				
342	126	/GOSUB*	EM	SLOW	0				
343	127	/DISPLAY*	ANSW14 .PIC-	SLOW	0	FADE			
344		/GOSUB*	PREG	0	0				
345		/IF*A	GOTO O15	0	0				
346		/IFNOT*A	GOTO T15	0	0				
347	O15	/DISPLAY*	MHG .PIC-	0	0	INSTAN			
348		/GOTO*	128	0	0				
349	T15	/DISPLAY*	TONTO1 .PIC-	0	0	INSTAN			
350		/DISPLAY*	TONTO2 .PIC-	0	0	INSTAN			

NO.	LABEL	COMMAND	Appendix A.2		TIME	WAIT	METHOD	SP
			PARAMETERS					
351		/GOTO*	127		0	0		
352	128	/DISPLAY*	ANSW15 .PIC-		SLOW	0	FADE	
353		/GOSUB*	PREG		0	0		
354		/IF*A	GOTO O16		0	0		
355		/IFNOT*A	GOTO T16		0	0		
356	O16	/DISPLAY*	MHG .PIC-		0	0	INSTAN	
357		/GOTO*	129		0	0		
358	T16	/DISPLAY*	TONTO1 .PIC-		0	0	INSTAN	
359		/DISPLAY*	TONTO2 .PIC-		0	0	INSTAN	
360		/GOTO*	128		0	0		
361	129	/DISPLAY*	ANSW16 .PIC-		SLOW	0	CRUSH	
362		/GOSUB*	PREG		0	0		
363		/IF*B	GOTO O17		0	0		
364		/IFNOT*B	GOTO T17		0	0		
365	O17	/DISPLAY*	MHG .PIC-		0	0	INSTAN	
366		/GOTO*	FINAL		0	0		
367	T17	/DISPLAY*	TONTO1 .PIC-		0	0	INSTAN	
368		/DISPLAY*	TONTO2 .PIC-		0	0	INSTAN	
369		/GOTO*	129		0	0		
370	00	/DISPLAY*	Z .PIC-		0	0	EXPLOD	
371		/BUTTON*1	1-436 103-478		0	0		
372		/BUTTON*2	539-436 640-477		0	0		
373		/INPUT*	1		0	0		
374		/RETURN*			0	0		
375	FINAL	/CLEAR*	16		0	0	EXPLOD	
376		/DISPLAY*	LISTO2 .PIC-		SLOW	0	EXPLOD	
377		/END*			0	0		
378		/*			0	0		
379	PREG	/BUTTON*A	70-363 184-456		0	0		
380		/BUTTON*B	272-362 397-453		0	0		
381		/BUTTON*C	510-362 635-454		0	0		
382		/INPUT*			0	0		
383		/RETURN*			0	0		
384	PM	/DISPLAY*	PREG .PIC-		0	0	INSTAN	
385		/CLEAR*	16		0	0	PUSH	
386		/RETURN*			0	0		
387		/*			0	0		
388		/*			0	0		
389		/*			0	0		
390		/*			0	0		
391		/*			0	0		
392		/*			0	0		
393		/*			0	0		
394		/*Story Last Line			0	0		



INDICE

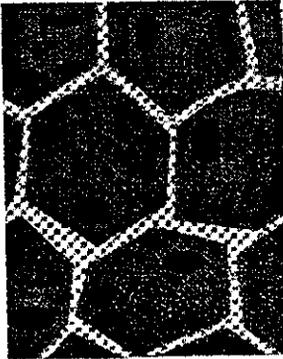
Para moverte a través de este tutorial utilizaras las siguientes señales:



Para ir a la pantalla **SIGUIENTE**



Para regresar a la pantalla **ANTERIOR**



La pared celular proporciona sostén y resistencia a los organismos vegetales.

Está compuesta químicamente por celulosa, sustancia dura, resistente e impermeable.

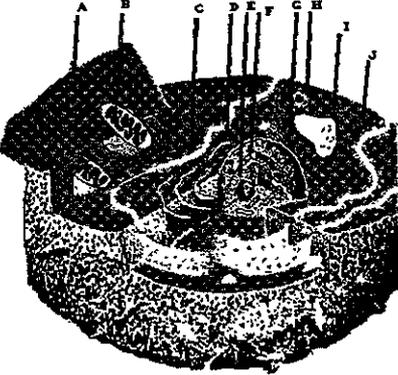


Diagram of a cell with labels A through J pointing to various organelles:

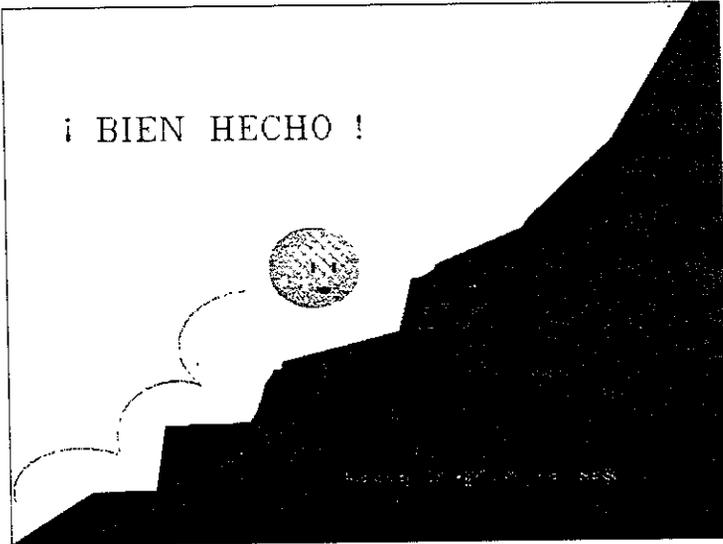
- A: Membran plasmatica.
- B: Mitochondria.
- C: Reticulo endoplasmatico.
- D: Nucleo.
- E: Nucleolo.
- F: Cromosoma.
- G: Vacuola.
- H: Lisosoma.
- I: Ribosoma.
- J: ...

Señala la letra del organelo que deseas consultar.

?

LOS LÍPIDOS SON

A GRASAS AZÚCARES C PROTEINAS



¡ALERTA!



ESA NO ES LA RESPUESTA
CORRECTA



¡RECTIFICA!

¡ M U Y



B I E N !

ES LA RESPUESTA

CORRECTA

ESCOGE UNA CLAVE CON TRES LETRAS ==>

MMH

COMO TE LLAMAS? ==>

JUANITO LOPEZ

CUANTOS AÑOS TIENES?

12

EMPIEZA TU CLASE DEL VIERNES 9/1/1998

BIENVENIDO

JUANITO LOPEZ

COMENZAMOS

EVALUACION	
RESPONDE CIERTO O FALSO (C / F)	RESPUESTA
Los lípidos se conocen comunmente como proteínas	C F
El adenosín difosfato está formado por tres radicales de fosfato	F F
Los postulados de la teoría celular son dos	F F
Las neuronas son células	C C
El ciclo celular dura 15 dias	C F

Las respuestas correctas se muestran en azul
PULSA ENTER PARA CONTINUAR

CALIFICACION DE LA LECCION

6

LO SIENTO

REPETIRAS LA LECCION

PRESIONA UNA TECLA PARA CONTINUAR

MANUAL DE INSTALACIÓN DEL
TUTORIAL CELLA.

Cree un directorio llamado "celda" en su disco duro

```
c:\> md celda
```

Cambie de directorio de trabajo a celda

```
c.\> cd celda
```

Copie el contenido de los discos del tutorial en el directorio celda

```
c:\celda> copy a:*.*
```

Cuando haya copiado el contenido de todos los discos teclee el nombre del programa ejecutable celula

```
c:\celda>celula
```

nota: la primera vez que teclee el programa el archivo de alumnos se reiniciara.

Con estos pasos usted ha instalado el tutorial celda en su computadora. A partir de aquí siga las instrucciones que dará la máquina ya dentro del tutorial.

Si desea la impresión del reporte de aprobación debe tener conectada la impresora cuando el alumno tome la lección 3, ya que después de la evaluación de esta lección se hace el promedio y se imprime en su reporte personal.