



9  
201

# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

HORARIOS DE PROFESORES  
FES ZARAGOZA.

## SEMINARIO DE INVESTIGACION INFORMÁTICA

QUE EN OPCIÓN AL GRADO DE  
LICENCIADO EN INFORMÁTICA  
PRESENTAN:  
JOSE ANTONIO GARCIA SANTOS  
LOURDES CAROLINA QUIJANO GONZALEZ

DIRECTOR DEL SEMINARIO:

L.A.E. MARIO NOVOA GAMAS



MEXICO, D. F.

1983

TESIS CON  
FALLA DE ORIGEN



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 está 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.

## **AGRADECIMIENTOS.**

**A la Universidad por contribuir a nuestra formación profesional y personal.**

**Al L.A.E. Mario Novoa Gamas por sus consejos y experiencias compartidas a lo largo de estos meses.**

**A nuestros padres y hermanos por el enorme apoyo en la conclusión de todas nuestras metas.**

**A Dios por todas sus bendiciones.**

# CONTENIDO

**Introducción.**

**Capítulo I: Situación Actual**

**Capítulo II: Análisis Estructurado**

- **Modelo Ambiental**
  - a) **Objetivo**
  - b) **Lista de Eventos**
  - c) **Diagrama de Contexto**
  
- **Modelo de Comportamiento**
  - a) **Diccionario de Datos**
  - b) **Diagramas de Flujo de Datos**
  - c) **Miniespecificaciones**

**Capítulo III: Diseño Estructurado**

- **Definición de Módulos**

**Capítulo IV: Implementación**

**Conclusiones**

**Bibliografía**

**Anexo**

## INDICE

Introducción.

<b>Capítulo I: Situación Actual .....</b>	<b>1</b>
<b>Capítulo II: Análisis Estructurado. ....</b>	<b>3</b>
• Modelo Ambiental .....	5
a) Objetivo.....	5
b) Lista de Eventos .....	6
c) Diagrama de Contexto .....	7
• Modelo de Comportamiento.....	10
a) Diccionario de Datos .....	10
b) Diagramas de Flujo de Datos .....	20
c) Miniespecificaciones.....	39
<b>Capítulo III: Diseño Estructurado.....</b>	<b>57</b>
• Definición de Módulos .....	61
<b>Capítulo IV: Implementación.....</b>	<b>77</b>
<b>Conclusiones .....</b>	<b>79</b>
<b>Bibliografía .....</b>	<b>82</b>
<b>Anexo .....</b>	<b>83</b>

## INTRODUCCION

Ante la necesidad de crear sistemas que cubran eficientemente los requerimientos de procesamiento de información en las organizaciones, a menores costos, se han creado metodologías y herramientas que contribuyan a satisfacer esta necesidad.

Por tanto, el sistema de Horarios de Profesores de la FES Zaragoza se desarrolló utilizando la metodología de Análisis y Diseño Estructurado\* de Edward Yourdon, la cual ofrece una combinación de herramientas gráficas y textuales de fácil comprensión que se caracteriza por su estilo sencillo y directo haciendo énfasis en las asignaturas esenciales del sistema y desechando o minimizando aquellos que son obvios e innecesarios para entender su funcionamiento.

En el Capítulo I se describe la fase de análisis estructurado con base en el "Modelo Esencial" que esquematiza el comportamiento interno del sistema y su relación con el ambiente.

En el Capítulo II se presenta el "Diseño Estructurado", en el cual a partir de los diagramas de flujo obtenidos en la fase de análisis, se define la estructura jerárquica de las funciones del sistema.

Por último, el Capítulo III "Implementación" presenta la codificación de instrucciones en el lenguaje elegido, en este caso Progress, para construir el sistema fundamentándolo en el análisis y diseño previamente desarrollados, incluyendo pruebas que garanticen su buen funcionamiento.

---

\* Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993.  
Structured Design. Yourdon, Edward. Ed. Prentice-Hall. USA, 1989.

## CAPITULO I

### Situación Actual

Actualmente la FES Zaragoza cuenta con un sistema de control manual de horarios y profesores, que consiste en registrar los movimientos en hojas tabulares por grupos y materias, sufriendo éstas continuas modificaciones que se derivan de los movimientos de alta, baja y licencia de los profesores. Enterando todos estos movimientos a la Administración de Personal a través de los formatos de propuesta.

Las bajas y licencias generan además, la elaboración del boletín de "Horas Vacantes" el cual detalla el motivo de la vacante, materia a impartir, requisitos a cubrir y funciones a desempeñar.

En caso de requerir información tal como:

- total de horas y grupos que tiene asignado un profesor
- total de profesores que imparten una asignatura
- control de horarios en que se imparte una asignatura

revisan todos los horarios y la elaboran manualmente, dando en la mayoría de las ocasiones una información poco confiable e inoportuna.

Además de esto, los usuarios tienen que enfrentar problemas como los siguientes:

- Los documentos no están bien organizados y en ocasiones se encuentran desactualizados, dificultando la búsqueda de información.
  - Cualquier modificación a los horarios o a los datos de los profesores implica tener que actualizar todos los documentos relacionados, por lo que su presentación se caracteriza por la combinación de escritura manual y a máquina.
-

Todas las labores anteriores implican una gran inversión de tiempo y esfuerzo humano, sin tener la certeza de que la necesidad de controlar los horarios y profesores se satisfaga con eficiencia.

Según cifras proporcionadas por el usuario, se manejan datos de aproximadamente 300 profesores y 900 horarios. Además, en el período interanual se deben elaborar los formatos de propuesta, horarios y boletines correspondientes a movimientos de altas, bajas, licencias y recontrataciones de profesores, sin contar los movimientos que ocurren en el transcurso del año.

Es por lo tanto, "**El Sistema de Horarios de Profesores**" el sistema que permitirá manejar los movimientos de asignación de grupos a profesores, emitir horarios, boletines de horas vacantes, formatos de propuesta y reportes confiables y oportunos, permitiendo una actualización fácil e integral.

## CAPITULO II

### Análisis Estructurado

En la fase de análisis estructurado definimos el objetivo del sistema, los elementos que lo componen y su interacción con el ambiente que lo rodea y que influye en su comportamiento.

Edward Yourdon define el objetivo de esta etapa de la siguiente manera:

"El propósito principal de la actividad de análisis es transformar las políticas del usuario y el esquema del proyecto, en una especificación estructurada." <sup>1</sup>

Para cumplir con el propósito de esta fase realizamos diversas entrevistas con los usuarios del sistema a fin de conocer detalladamente las funciones que realiza y las áreas con las que interactúa, esta información la modelamos a través de las herramientas gráficas y textuales, que nos facilitaron:

- La comunicación con el usuario.
- Resaltar aspectos importantes del sistema y descartar los irrelevantes.
- Realizar correcciones y cambios a las especificaciones a bajo costo y sin riesgo alguno.
- Asegurar que el modelo representará exactamente lo que el usuario requería.

Y fueron la guía para la construcción del sistema.

---

<sup>1</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana, México, 1993. P.101

El modelo que utilizamos en el análisis del sistema es el Modelo Esencial, definido como "lo que el sistema debe hacer para satisfacer los requerimientos del usuario diciendo lo mínimo posible a cerca de cómo se implantará"<sup>2</sup>.

Este se subdivide a su vez en dos modelos:

- El *Modelo Ambiental*.
- El *Modelo de Comportamiento*.

Ambos modelos se describen en las páginas siguientes.

---

<sup>2</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 358

### **Modelo Ambiental.**

El definir los límites del sistema fue uno de los problemas a los que nos enfrentamos al iniciar el análisis, debido a la importancia de determinar los elementos que lo conforman y cuales estan fuera de él.

El Modelo Ambiental, al definir "las interfaces entre el sistema y el resto del universo, es decir, el ambiente"<sup>3</sup> nos brindó las herramientas necesarias para delimitarlo.

El modelo ambiental se constituye por tres partes fundamentales:

#### **a) Objetivo**

El Objetivo de un sistema "es una declaración textual breve y concisa del propósito del sistema"<sup>4</sup>. Por estar dirigida al usuario, niveles directivos y otras personas no involucradas en el desarrollo del sistema debe de estar definido utilizando un lenguaje sencillo y fácil de entender.

El objetivo del sistema lo definimos conjuntamente con el usuario, realizamos varias revisiones hasta estar de acuerdo en que el objetivo definía claramente las funciones principales del sistema.

Cabe señalar que al definir el objetivo enfatizamos sólo los elementos relevantes del sistema, aquellos comunes a todo sistema y que, por lo tanto, no son indispensables para definirlo particularmente, fueron descartados.

---

<sup>3</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 369

<sup>4</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 373

El objetivo del Sistema de Horarios de Profesores es:

***Manejar los movimientos de asignación de grupos a profesores, emitir horarios, boletines de horas vacantes y formatos de propuesta.***

b) *Lista de Eventos*

La Lista de Eventos se define como "una lista narrativa de los estímulos que ocurren en el mundo exterior a los cuales el sistema debe responder."

Para su redacción fue necesario definir, conjuntamente con los usuarios, los acontecimientos o acciones relevantes de los terminadores<sup>5</sup> que provocaran una reacción en el sistema, como la emisión de documentos impresos y modificaciones en los datos almacenados.

En el análisis del Sistema de Horarios de Profesores se identificaron los siguientes eventos:

- 1. La Coordinación de Ciclo ingresa movimiento.***
- 2. La Coordinación de Ciclo ingresa movimiento de profesores.***
- 3. La Coordinación de Ciclo ingresa movimiento de horarios.***
- 4. La Coordinación de Ciclo solicita horarios.***
- 5. La Coordinación de Ciclo solicita boletín de horas vacantes.***
- 6. La Administración de Personal solicita formato de propuesta.***

Paralelamente a esta lista de eventos, elaboramos el diagrama de contexto, que se describe a continuación.

---

<sup>5</sup> Personas, organizaciones o sistemas con los que se comunica nuestro sistema.

c) *Diagrama de contexto:*

El Diagrama de Contexto "es un caso especial de diagrama de flujo de datos en donde una sola burbuja representa a todo el sistema"<sup>6</sup>. En este diagrama se muestra la interacción del sistema con el mundo exterior. Dando especial importancia a:

- Los *terminadores*, que son las "personas, organizaciones y sistemas con los que se comunica el sistema."<sup>7</sup>
- Los *datos recibidos por el sistema* que provocan alguna reacción o respuesta por parte del mismo.
- Los *datos producidos por el sistema* que son enviados al exterior.
- Los *almacenes de datos* que el sistema comparte con los terminadores.
- Los *límites* entre el sistema y el mundo exterior.

En la elaboración del Diagrama de Contexto del Sistema de Horarios de Profesores cuidamos que hubiera consistencia con la información presentada en la Lista de Eventos. Esta consistencia garantiza que las transacciones que los usuarios mandan al sistema reciban la respuesta requerida.

---

<sup>6</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 374

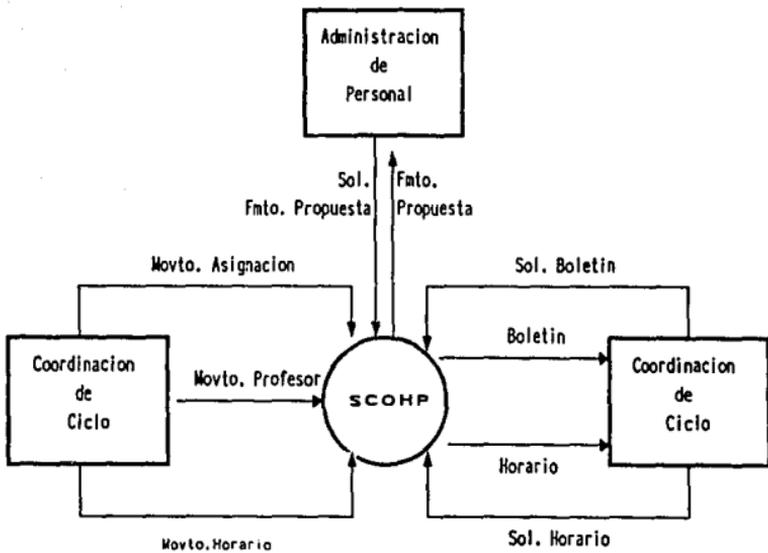
<sup>7</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 374

Las convenciones utilizadas en la elaboración del diagrama de contexto son las siguientes:

<i><b>Símbolo</b></i>	<i><b>Significado</b></i>
	Representa al sistema y contiene su nombre en el interior
	Representa a un terminador y contiene su nombre en el interior
	Representa un flujo de datos y su dirección (hacia adentro o hacia afuera del sistema). El nombre del flujo de datos se presenta junto a la flecha correspondiente

El diagrama de contexto del sistema de Horarios de Profesores se presenta a continuación.

DIAGRAMA DE CONTEXTO



## Modelo de Comportamiento.

El modelo de comportamiento, como su nombre lo indica, "describe el comportamiento que del sistema se requiere para que interactúe de manera exitosa con el ambiente"<sup>8</sup>. En él se define la composición de los datos utilizados por el sistema, los flujos de datos a través de los procesos y las operaciones realizadas en cada uno de los procesos.

En el Sistema de Horarios de Profesores, a través de este modelo, una vez definido el alcance y límite del sistema especificamos detalladamente los datos que manejaría y las funciones a realizar para cumplir con el objetivo previamente definido.

Las herramientas utilizadas en este modelo son<sup>9</sup>:

- Diccionario de Datos
- Diagramas de Flujo de Datos (DFD)
- Miniespecificaciones

### a) *Diccionario de Datos*

Es la herramienta que utilizamos para describir los elementos datos declarados en los diagramas de flujo de datos; el significado de los flujos y almacenamientos, su composición, y las unidades y valores de piezas elementales de información.

"El diccionario de datos es un listado organizado de todos los datos pertinentes al sistema, con definiciones precisas y rigurosas para que tanto

---

<sup>8</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 360

<sup>9</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 361

el usuario como el analista tengan un entendimiento común de todas las entradas, salidas, asignaturas de almacenes y cálculos intermedios."<sup>10</sup>

Con el fin de asegurar una completa comprensión por parte del usuario, el diccionario de datos del sistema de Horarios de Profesores se elaboró respetando los términos manejados en él por el usuario, y para ello fue indispensable su colaboración.

A continuación se presenta el diccionario de datos del sistema de Horarios de Profesores y las convenciones que fueron utilizadas para su redacción.

---

<sup>10</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 212

## **DICCIONARIO DE DATOS**

### **Convenciones utilizadas**

- =** Está integrado por
- +** Más
- { }** Iteración
- [ ]** Seleccionar una de las opciones
- |** Separador de opciones
- &** Espacio en blanco.

DICCIONARIO DE DATOS

Alta =	
Alta-Hor =	* Asignación de un profesor a un grupo.
Alta-prof =	* Dar de alta los datos de un nuevo horario.
	* Dar de alta los datos personales y laborales de un profesor.
Ap-materno =	1{caracter-legal}
Ap-paterno =	1{caracter-legal}
Asignatura =	1{Caracter-legal}30
	* Nombre de la asignatura a impartirse o asignarse.
Ayudante =	
	* Profesor auxiliar en la impartición de un asignatura.
Baja =	
	* Dar de baja las horas asignadas de un grupo a un profesor.
Baja-Hor =	
	* Dar de baja los datos de un horario.
Baja-prof =	
	* Dar de baja los datos laborales y personales de un profesor.

**HORARIOS DE PROFESORES**  
Análisis Estructurado

---

Calle =	{[A - Z a - z 0 - 9 '  & ]} 26 * Nombre de la calle y número del domicilio del profesor.
Cambio-Hor =	* Modificar los datos de un horario.
Cambio-Prof =	* Modificar los datos laborales y/o personales de un profesor.
Carácter-legal =	[A - Z a - z 0 - 9 '  ]
Categoría =	{1A = Ayudante A  1B = Ayudante B  2A = Asignatura A   2B=Asignatura B  3A = Asignatura A definitivo  3B = Asignatura B definitivo  * Categoría laboral con la que el profesor es asignado a un grupo.
Causa-baja =	{caracter-legal} * Motivo por el cual se realiza la baja de un profesor.
Causa-movto =	{Caracter-legal} * Motivo por el cual se realiza el movimiento de asignación.
Ciclo =	{1   2   3 } * Agrupación de semestres en la que se ubica un asignatura.

**HORARIOS DE PROFESORES**  
Análisis Estructurado

---

Ciudad =	{Caracter-legal} * Ciudad del domicilio del profesor.
Clínica =	{Caracter-legal} * Nombre de la clínica donde se impartirá un asignatura.
Cod-Programatico =	4{0 - 9}4 * Clave asignada a cada combinación de grupo y asignatura.
Colonia =	{Caracter-legal} * Colonia del domicilio del profesor.
CP =	5{0 - 9}5 * Código postal del domicilio del profesor.
Cve-Movto =	{0 - 9} * Número consecutivo asignado a cada movimiento de asignación de grupos a profesores.
Día-Hora =	[Lu  Ma  Mi  Ju  Vi ] + [7 - 21] + [8 - 22] * Día y horas en las que se impartirá un asignatura.
Dígito =	{0 - 9}
Entidad =	{Caracter-legal} * Ciudad del domicilio del profesor.

**HORARIOS DE PROFESORES**  
Análisis Estructurado

---

Escolaridad =	[Licenciatura  Especialidad  Maestría  Doctorado ] + {Caracter-legal} * Grado de escolaridad alcanzado por el profesor + el área.
Fecha-baja =	6{0 - 9}6 * Fecha en que se da de baja un profesor con formato dd/mm/yy.
Fecha-ingreso =	6{0 - 9}6 * Fecha de ingreso de un profesor a la FESZ con formato dd/mm/yy.
Fecha-movto =	6{0 - 9}6 * Fecha en que se registra un movimiento con formato dd/mm/yy.
Función =	[1 = Titular  2 = Ayudante ] * Función con la que un profesor es asignado a un grupo.
Grupo =	4{0 - 9}4 * Clave del grupo con formato numérico.
Hrs-practica =	1{1 - 7}1 * Horas impartidas de práctica en un asignatura.
Hrs-teoría =	1{1 - 7}1 * Horas impartidas de teoría en un asignatura.

**HORARIOS DE PROFESORES**  
Análisis Estructurado

---

Licencia =	* Dar de baja temporalmente las hora(s) asignadas de un grupo(s) a un profesor.
Lugar =	[salón  clínica ] * Salón o clínica en que se impartirá un asignatura.
Movto =	* Datos para realizar un alta, baja o licencia en la asignación de un grupo a un profesor.
Movto-Horario =	* Datos para realizar una alta, baja o cambio en los horarios.
Movto-Prof =	* Datos para realizar una alta, baja o cambio en los datos de los profesores.
Nacionalidad =	1{Caracter-legal }15 * Nacionalidad del profesor.
Nombre =	1{caracter-legal}
Nombre-prof =	Ap-paterno + Ap-materno + Nombre(s) * Nombre del profesor con formato.
Num-boletín =	1{0 - 9}

## HORARIOS DE PROFESORES

### Análisis Estructurado

---

\* Número consecutivo asignado por el sistema para referenciar los datos del boletín.

RFC =	$4\{A - Z\} + 6\{0 - 9\}6$ * Registro Federal de contribuyentes del profesor.
Salón =	$\{A - E\} + 1\{0 - 9\}3$ * Aula en que se impartirá un asignatura.
Semestre =	$\{0 - 9\}2$ * Semestre en que se imparte la asignatura.
Sexo =	$\{F   M\}$ * Sexo del profesor.
Sol-boletín =	* Datos para emitir un boletín de horas vacantes.
Sol-Fmto-Propuesta =	* Datos para emitir un formato de propuesta.
Sol-Horario =	* Datos para emitir un horario.
Teléfono =	$7\{0 - 9\}7$ * Teléfono del profesor.
Tipo-Horario =	* Tipo de horario que se emitirá.

## HORARIOS DE PROFESORES

### Análisis Estructurado

---

- Tipo-movto-horario = [Alta-hor |Baja-hor |Cambio-Hor ]  
\* Tipo de movimiento a realizarse en los horarios.
- Tipo-movto = [Alta |Baja |Licencia ]  
\* Tipo de movimiento a realizarse en la asignación de grupos a profesores.
- Tipo-movto-prof = [Alta-prof |Baja-prof |Cambio-prof ]  
\* Movimiento a realizarse en los datos del profesor.
- Titular =  
\* Profesor responsable de impartir un asignatura.
- Vigencia = 6{0 - 9}  
\* Fecha a partir de la cual tiene validez la asignación de un grupo a un profesor con formato dd/mm/yy.

b) *Diagrama de Flujo de Datos (DFD)*

Los diagramas de flujo de datos describen los procesos que realiza el sistema, las entradas que recibe y las salidas que proporciona. Es una herramienta que permite visualizar un sistema como una red de procesos funcionales, conectados entre sí por flujos de almacenamientos de datos."<sup>11</sup>

Es por ello que se dice que el DFD "proporciona un punto de vista un sistema, el orientado a las funciones."<sup>12</sup>

Para evitar que la representación de un sistema por medio de un DFD se vuelva demasiado compleja se debe "organizar el DFD global en una serie de niveles de modo que cada uno proporcione sucesivamente más detalles sobre una porción del nivel anterior."<sup>13</sup>

Por tanto es recomendable, la creación de diagramas de flujo a partir de un nivel 0, que muestre una vista global del sistema a través de burbujas que representen las actividades identificadas en el modelo ambiental, flujos de datos que muestran el movimiento de datos y almacenamientos que representan los datos que el sistema debe conservar.

A partir de este nivel, se hará un nuevo diagrama por cada círculo que requiera de mayor explicación.

---

<sup>11</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993 P. 396.

<sup>12</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993 P. 396.

<sup>13</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 186

Los procesos definidos en cada DFD fueron numerados para facilitar el seguimiento de los procesos a través de los diferentes niveles de diagramación. Por ejemplo, en el diagrama 0, al proceso llamado "Procesa Asignación" se le asignó el número 1. En el siguiente nivel fue necesario elaborar un nuevo DFD para detallar el proceso número 1. Por lo tanto, los procesos de este nuevo DFD se numeraron como 1.1, 1.2, y así sucesivamente.

Es importante mencionar que el número de cada proceso no significa que éste se deba ejecutar en ese orden.

Los DFD del sistema de Horarios de Profesores se presenta en las siguientes páginas.

## DIAGRAMAS DE FLUJO DE DATOS

Convenciones utilizadas:

<b><i>Símbolo</i></b>	<b><i>Significado</i></b>
	<b>Proceso</b> Representa una parte del sistema donde los datos de entrada sufren una modificación y se transforman en salidas
	<b>Flujo</b> Muestra la dirección en que se mueve la información.
	<b>Almacén</b> Representa un paquete de datos en reposo (como discos, cintas y otros tipos de dispositivos de almacenamiento de datos)

HORARIOS DE PROFESORES  
Análisis Estructurado

DIAGRAMA DE FLUJO DE DATOS

DIAGRAMA 0.

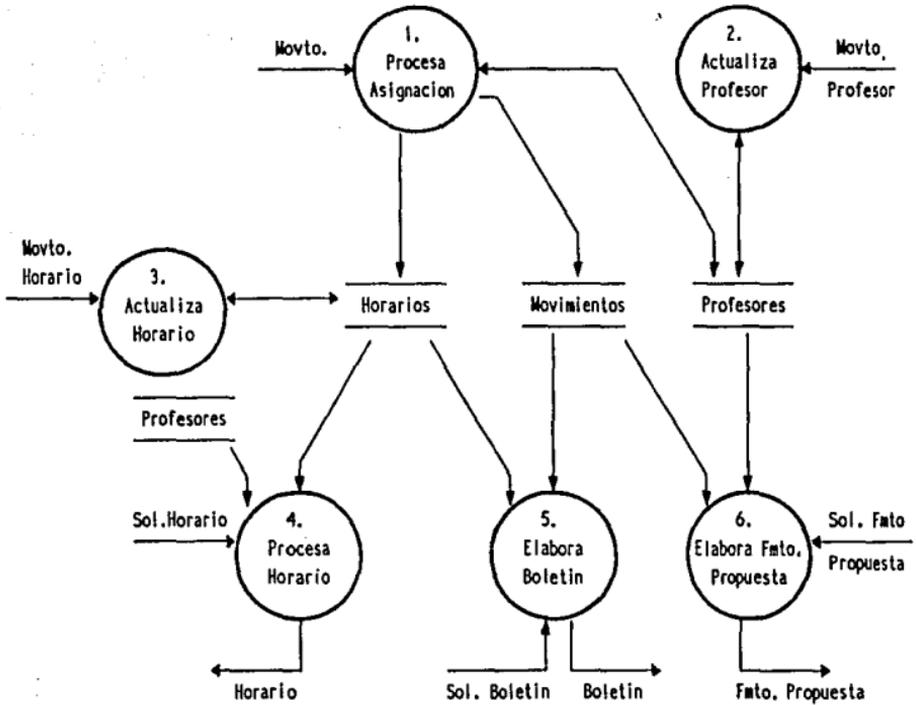
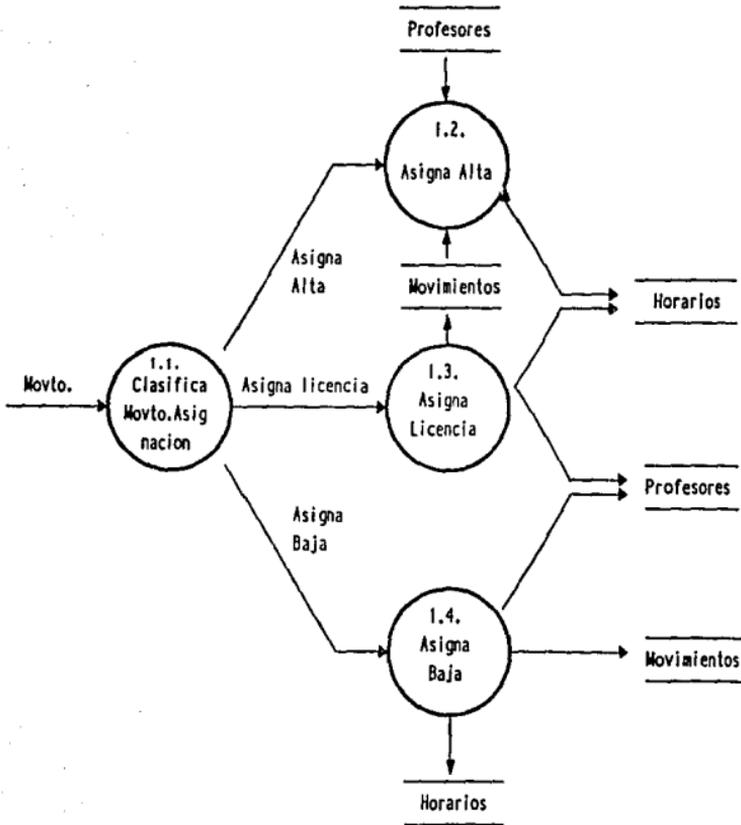


DIAGRAMA DE FLUJO DE DATOS

1.0. Proceso de Asignación



---

DIAGRAMA DE FLUJO DE DATOS

1.2. Asigna Alta



DIAGRAMA DE FLUJO DE DATOS

1.3. Asigna licencia



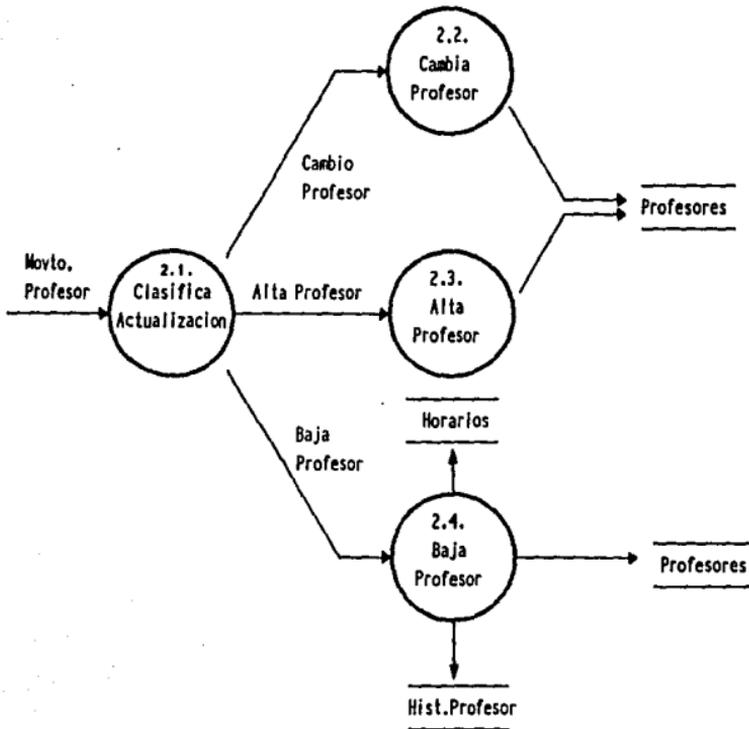
DIAGRAMA DE FLUJO DE DATOS

1.4. Asigna baja



DIAGRAMA DE FLUJO DE DATOS

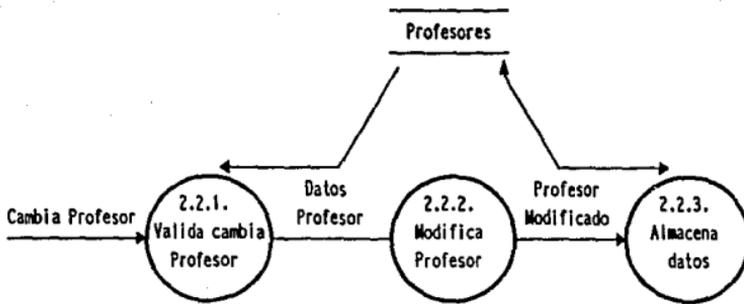
2.0. Actualiza Profesor



---

DIAGRAMA DE FLUJO DE DATOS

2.2. Cambia Profesor



---

DIAGRAMA DE FLUJO DE DATOS

2.3. Alta Profesor

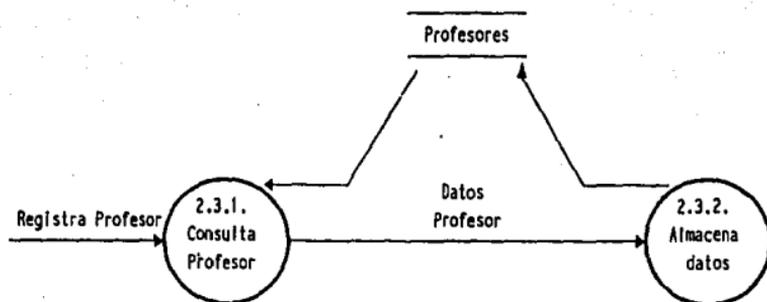


DIAGRAMA DE FLUJO DE DATOS

2.4. Baja Profesor

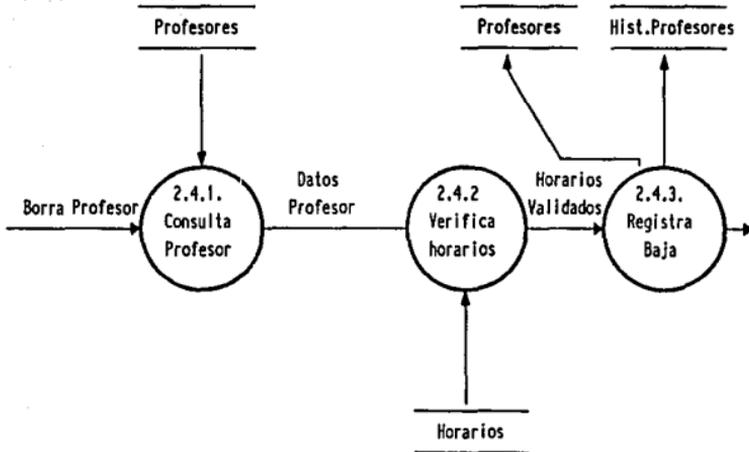


DIAGRAMA DE FLUJO DE DATOS

3.0. Actualiza Horarios

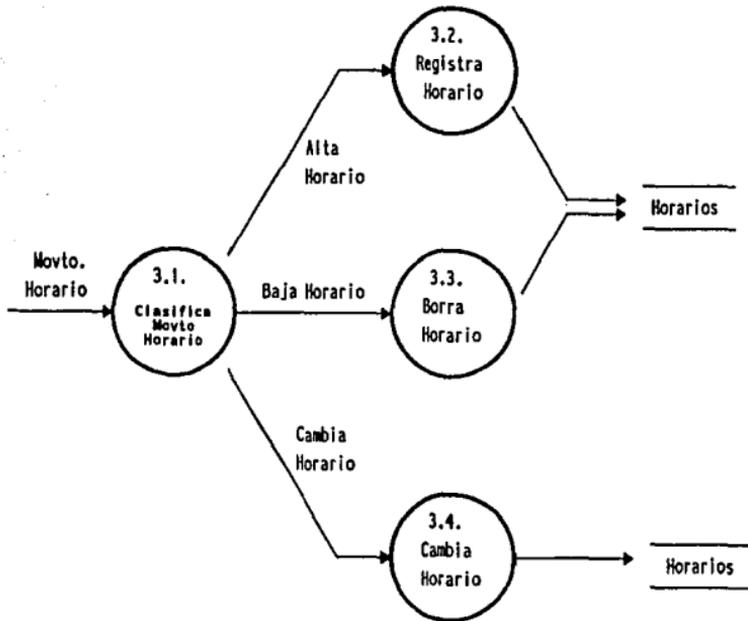


DIAGRAMA DE FLUJO DE DATOS

3.2. Registra Horario

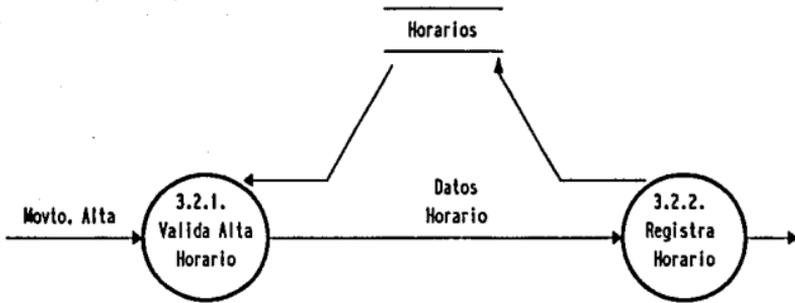


DIAGRAMA DE FLUJO DE DATOS

3.3. Borra Horario

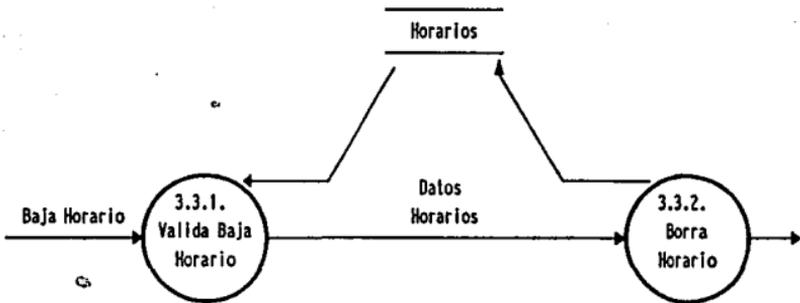


DIAGRAMA DE FLUJO DE DATOS

3.4. Cambia Horario

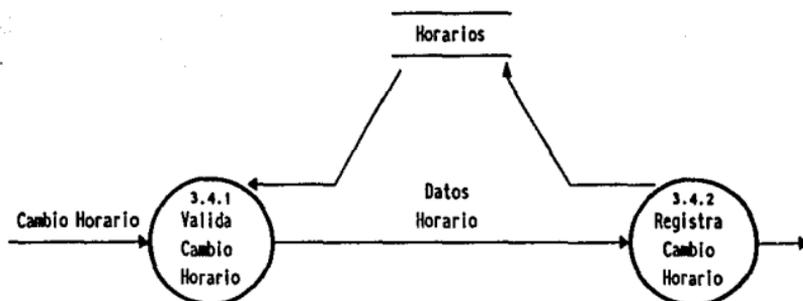


DIAGRAMA DE FLUJO DE DATOS

4.0. Procesa Horario

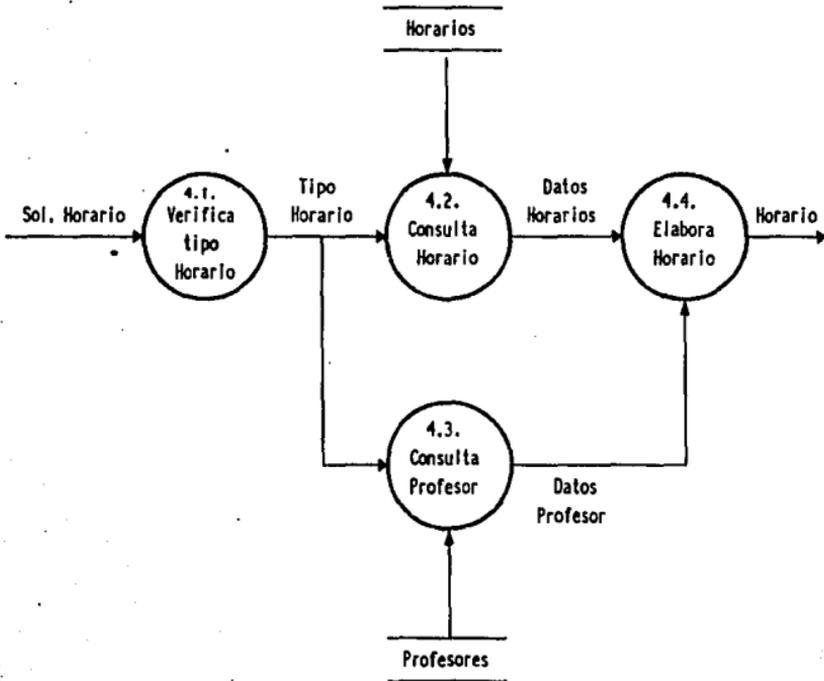
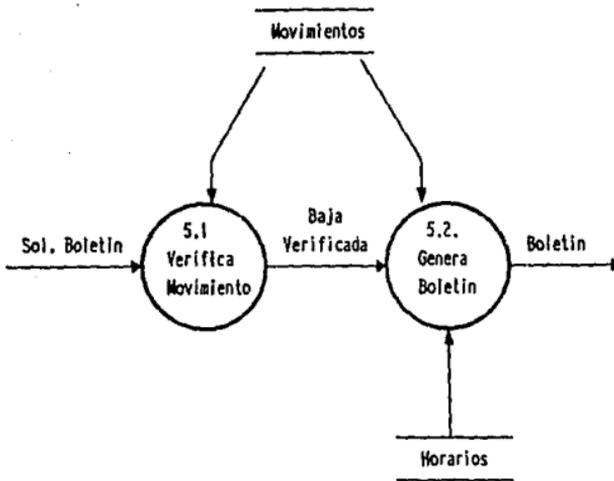


DIAGRAMA DE FLUJO DE DATOS

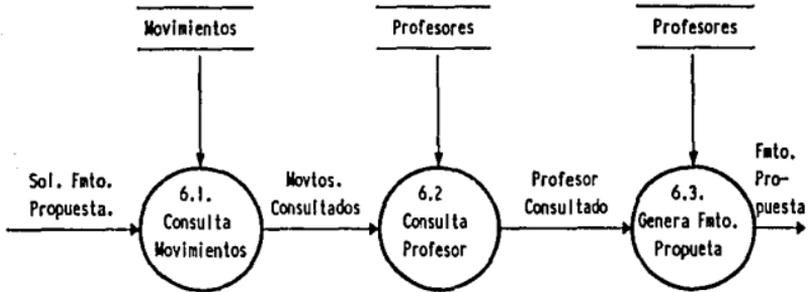
5.0. Elabora Boletín



---

DIAGRAMA DE FLUJO DE DATOS

6.0. Elabora Fmto. Propuesta



c) *Miniespecificaciones.*

A través de su redacción obtuvimos la descripción detallada de las actividades definidas en los diagramas de flujo de datos de nivel más bajo.

Esta etapa se realiza sin considerar el lenguaje de programación que se utilizará, define "lo que debe hacerse para transformar entradas en salidas"<sup>14</sup> y constituye la base para elaborar la programación del sistema, apoyada con el diseño.

Las miniespecificaciones se construyeron en *lenguaje estructurado*, al combinar el idioma cotidiano (en este caso el español) con la estructura del lenguaje formal de programación.

Entre las ventajas de utilizar un lenguaje estructurado para definir las especificaciones del proceso, se encuentran las siguientes:

El uso de palabras y frases del idioma cotidiano hace que la comprensión de las miniespecificaciones sea más accesible para el usuario.

El uso de la estructura del lenguaje formal de programación contribuye a evitar especificaciones demasiado detalladas y confusas, al definir un número limitado de palabras y reglas que se van a utilizar.

El uso de los términos definidos en el diccionario de datos hace que las miniespecificaciones sean más fáciles de entender porque limita el lenguaje utilizado en las mismas, además de hacer que haya más consistencia en el proceso y documentación del análisis.

---

<sup>14</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 227

Las convenciones utilizadas en la elaboración de las miniespecificaciones del sistema de Horarios de Profesores son las siguientes:

### MINIESPECIFICACIONES

#### Convenciones Utilizadas.

" "	Contiene el mensaje a desplegar en pantalla
! * ?	Contiene el nombre del proceso
<i>Itálicas</i>	Nombre de campo (dato de un archivo)
v-<variable>	Nombre de variable de trabajo
<b>MAYUSCULAS</b>	Nombre de archivo
<b>Borra registro</b>	Elimina registro previamente localizado en un archivo
<b>Busca campo Donde</b> <condición>	Localiza en un archivo el primer registro que cumpla con la condición especificada
<b>Despliega</b>	Manda un mensaje o datos a pantalla
<b>Define</b>	Especifica las variables de trabajo o los reportes que se van a utilizar

**En caso de:**

<condicion-1>, Haz:

<instrucciones>

<condicion-2>, Haz:

<instrucciones>

:

<condicion-n>, Haz:

<instrucciones>

<b>Fin-En caso</b>	Ejecuta las instrucciones asociadas a la condición que se cumple
<b>Envía</b>	Manda datos a un reporte que se imprimirá posteriormente
<b>Escribe</b>	Crea un nuevo registro en un archivo
<b>Mientras &lt;condición&gt; Haz:</b>	
<instrucciones>	
<b>Fin-Mientras</b>	Realiza las instrucciones especificadas mientras la condición se cumpla
<b>Imprime</b>	Envía un reporte a impresora
<b>Modifica</b>	Permite la modificación de datos en pantalla que posteriormente se actualizarán en un archivo
<b>Obtiene</b>	Toma de la pantalla <sup>15</sup> datos proporcionados por el usuario

---

<sup>15</sup> Se utilizó el término "pantalla" para hacer las explicaciones más fáciles de entender, tomando en cuenta que para los usuarios es un poco difícil asimilar la idea de que la información entra por el teclado y no por la pantalla.

<b>Reescribe</b>	Actualiza registro(s) ya existentes en un archivo
<b>Si &lt;condición&gt; Entonces</b> <instrucciones>	
<b>Sino &lt;instrucciones&gt;</b>	
<b>Fin-Si</b>	Ejecuta el primer bloque de instrucciones si se cumple la condición. En caso contrario ejecuta el segundo bloque.

### MINIESPECIFICACIONES

**Define** variable v-opcion.

**Obtiene** v-opcion

**En caso de que:**

v-opcion = 1, **Haz:** [1. Procesa asignacion]

v-opcion = 2, **Haz:** [2. Actualiza profesor]

v-opcion = 3, **Haz:** [3. Actualiza horario]

v-opcion = 4, **Haz:** [4. Procesa horario]

v-opcion = 5, **Haz:** [5. Elabora boletin]

v-opcion = 6, **Haz:** [6. Elabora fmo. Propuesta]

**Fin-En caso**

## MINIESPECIFICACIONES

### 1. Procesa Asignacion

**/\* 1.1. Clasifica movimiento de asignacion \*/**

**Define variable v-opcion**

**Obtiene v-opcion**

**En caso de que:**

v-opcion = 1, Haz: [1.2. Asigna alta]

v-opcion = 2, Haz: [1.3. Asigna licencia]

v-opcion = 3, Haz: [1.4. Asigna baja]

**Fin-En caso**

### 1.2. Asigna Alta

**Define variable v-rfc, v-cod-prog, v-no-movto, vcve-fun**

**/\* 1.2.1. Valida profesor \*/**

**Obtiene v-rfc**

**Busca rfc en PROFESOR Donde PROFESOR.rfc = v-rfc**

**Si encuentra registro Entonces Haz:**

**/\* 1.2.2. Valida-horarios \*/**

**Obtiene v-cod-prog vcve-fun**

**Busca cod-prog en HORARIOS Donde HORARIOS.cod-prog =  
v-cod-prog y HORARIOS.cve-fun = vcve-fun**

**MINIESPECIFICACIONES**

**Si encuentra registro Entonces Haz:**

**Si HORARIO.rfc = desocupado Entonces Haz:**

**/\* 1.2.3 Registra movto-alta \*/**

**Reescribe registro en HORARIOS**

**Escribe registro en MOVIMIENTOS**

**Sino**

**Despliega "El Horario ya esta asignado"**

**Fin-Si**

**Sino**

**Despliega "El Horario no esta registrado"**

**Fin-Si**

**Sino**

**Despliega "El Profesor no esta registrado"**

**Fin-Si**

**1.3. Asigna Licencia**

**Define variable v-rc, v-cod-prog, v-num-movto, vcve-fun**

**/\* 1.3.1. Valida Profesor \*/**

**Obtiene v-rc**

**Busca rfc en PROFESOR Donde PROFESOR.rfc = v-rc**

**MINIESPECIFICACIONES**

**Si encuentra registro Entonces Haz:**

*/\* 1.3.2. Valida horarios \*/*

**Obtiene** *vcod-prog vcve-fun*

**Busca** *cod-prog* en HORARIO Donde HORARIOS.*cod-prog* =  
*v-cod-prog* y HORARIO.*rfc* = *v-rfc* y HORARIOS.*cve-fun* =  
*vcve-fun*

**Si encuentra registro Entonces Haz:**

*/\* 1.3.3. Registra movto-lic \*/*

**Graba** registro en HORARIO

**Graba** registro en MOVIMIENTOS

**Sino**

**Despliega** "El Profesor no esta asignado a esta materia"

**Fin-Si**

**Sino**

**Despliega** "El Profesor no esta registrado"

**Fin-Si**

**1.4. Asigna Baja**

**Define** variable *v-rfc, v-cod-prog, v-num-movto, vcve-fun*

*/\* 1.4.1. Valida profesor \*/*

**Busca** *rfc* en PROFESOR Donde PROFESOR.*rfc* = *v-rfc*

**MINIESPECIFICACIONES**

**Si encuentra registro Entonces Haz:**

**/\* 1.4.2. Valida Horarios \*/**

**Busca cod-prog en HORARIOS Donde HORARIOS.cod-prog =  
v-cod-prog y HORARIOS.rfc = v-rfc y HORARIOS.cve-fun  
= vcve-fun**

**Si encuentra registro Entonces Haz:**

**/\* 1.4.3. Registra movto.baja \*/**

**Graba registro en HORARIOS**

**Graba registro en MOVIMIENTOS**

**Sino**

**Despliega "El Profesor no esta asignado a esta  
asignatura"**

**Fin-Si**

**Sino**

**Despliega "El Profesor no esta registrado"**

**Fin-Si**

## MINIESPECIFICACIONES

### 2. Actualiza Profesor

**/\* 2.1. Clasifica actualizacion \*/**

**Define** var v-opcion

**Obtiene** v-opcion

**En caso de que:**

v-opcion = 1, **Haz:** [2.2. Cambia Profesor]

v-opcion = 2, **Haz:** [2.3. Alta Profesor]

v-opcion = 3, **Haz:** [2.4. Baja Profesor]

**Fin-En caso**

### 2.2. Cambia Profesor

**/\* 2.2.1. Valida Cambio Profesor \*/**

**Define** v-rfc

**Obtiene** v-rfc

**Busca** rfc en PROFESOR **Donde** PROFESOR.rfc = v-rfc

**Si** encuentra registro **Entonces Haz:**

**/\* 2.2.2. Modifica Profesor \*/**

**Modifica** Registro PROFESOR

**/\* 2.2.3. Almacena datos \*/**

**Reescribe** registro en PROFESOR

**Sino**

**Despliega** "El Profesor no se encuentra registrado"

**Fin-Si**

## MINIESPECIFICACIONES

### 2.3. Registra Profesor

*/\* 2.3.1. Consulta Profesor \*/*

**Define** variable v-rfc

**Obtiene** v-rfc

**Busca** rfc en PROFESOR **Donde** PROFESOR.rfc = v-rfc

**Si** encuentra registro **Entonces Haz:**

*/\* 2.3.2. Almacena datos \*/*

**Escribe** registro en PROFESOR

**Fin-Si**

**Si** encuentra

**Despliega** "El Profesor ya se encuentra registrado"

**Fin-Si**

### 2.4. Baja Profesor

*/\* 2.4.1. Consulta Profesor \*/*

**Define** variable v-rfc, v-num-movlo

**Obtiene** v-rfc

**Busca** rfc en PROFESOR **Donde** PROFESOR.rfc = v-rfc

**Si** encuentra registro **Entonces Haz:**

*/\* 2.4.2. Verifica horarios \*/*

**Busca** rfc en HORARIOS **Donde** HORARIOS.rfc = v-rfc

**MINIESPECIFICACIONES**

**Si no encuentra registro Entonces Haz:**

**/\* 2.4.3. Registra Baja \*/**

**Escribe registro en HISTORICO**

**Borra registro en PROFESOR**

**Sino**

**Despliega "El profesor tiene grupos asignados, no se puede Borrar".**

**Fin-Si**

**Sino,**

**Despliega "El Profesor no se encuentra registrado"**

**Fin-Si**

## MINIESPECIFICACIONES

### 3. Actualiza Horario

/\* 3.1. Clasifica Movto-horario \*/

**Define** variable v-opcion

**Obtiene** v-opcion

**En caso de que:**

v-opcion = 1, Haz: [3.2. Registra horario]

v-opcion = 2, Haz: [3.3. Borra horario]

v-opcion = 3, Haz: [3.4 Cambia horario ]

**Fin-En caso**

### 3.2. Registra Horario

/\* 3.2.1. Valida alta horario \*/

**Define** variable v-cod-prog, vcve-fun

**Obtiene** v-cod-prog, vcve-fun

**Busca** cod-prog en HORARIOS Donde HORARIOS.cod-prog =

v-cod-prog y HORARIOS.cve-func = vcve-fun

**Si** no encuentra registro **Entonces** Haz:

/\* 3.2.2. Registra Horario \*/

**Escribe** registro en HORARIOS

**Sino**

**Despliega** "El Horario ya se encuentra registrado"

**Fin-Si**

### MINIESPECIFICACIONES

#### 3.3. Borra Horario

*/\* 3.3.1. Valida Baja Horario \*/*

**Define** variable v-cod-prog, vcve-fun

**Obtiene** v-cod-prog, vcve-fun

**Busca** cod-prog en HORARIOS Donde HORARIOS.cod-prog =  
v-cod-prog y HORARIOS.rfc = vacio y HORARIO.cve-fun =  
vcve-fun

**Si** encuentra registro **Entonces Haz:**

*/\* 3.3.3. Borra Horario \*/*

**Borra** registro en HORARIOS

**Sino**

**Despliega** "El Horario no se puede Borrar, ya que tiene asignado profesor".

**Fin-Si**

#### 3.4. Cambia Horario

*/\* 3.4.1. Valida Cambio Horario \*/*

**Define** variable v-cod-prog, vcve-fun

**Obtiene** v-cod-prog, vcve-fun

**Busca** cod-prog en HORARIOS Donde HORARIOS.cod-prog =  
v-cod-prog y HORARIOS.cve-fun = vcve-fun

**MINIESPECIFICACIONES**

**Si encuentra registro Entonces Haz:**

**/\* 3.4.2. Registra cambio Horario \*/**

**Reescribe registro en HORARIO**

**Sino**

**Despliega "El Horario no existe "**

**Fin-Si**

MINIESPECIFICACIONES

4. Procesa Horario

/\* 4.1. Verifica tipo horario \*/

Define variable v-tipo-horario, v-rfc

Define reporte Rep.Horarios

Obtiene v-tipo-horario

/\* 4.2. Consulta Horario \*/

Obtiene v-rfc

Mientras haya registro en HORARIOS Haz:

Busca rfc en HORARIOS Donde HORARIOS.rf = v-rfc

Si encuentra registro Entonces Haz:

Graba registro HORARIOS en Rep.Horarios

Fin-Mientras.

/\* 4.3. Consulta Profesor \*/

Busca rfc en PROFESOR Donde PROFESOR.rfc = v-rfc

Si encuentra registro Entonces Haz:

Envía datos PROFESOR a Rep.Horario

/\* 4.4 Elabora horario \*/

Imprime Rep.horario.

Fin-Si

## MINIESPECIFICACIONES

### 5. Elabora Boletín

*/\* 5.1. Verifica movimiento \*/*

**Define** variable v-num-movto, vno-boletin, vcod-prog

**Define** variable vcve-fun

**Define** Reporte Rep.Boletín

**Obtiene** v-num-movto, vcod-prog

**Busca** num.movto en MOVIMIENTOS Donde

MOVIMIENTOS.num.movto = v-num-movto y

MOVIMIENTOS.cod-prog = vcod-prog

**Si** encuentra registro **Entonces Haz:**

*/\* 5.2. Genera Boletín \*/*

**Busca** cod-prog en HORARIOS Donde HORARIOS.cod-prog =

MOVIMIENTOS.cod-prog y HORARIOS.cve-fun =

MOVIMIENTOS.cve-fun

**Si** encuentra registro **Entonces Haz:**

**Imprime** Rep-boletín

**Sino**

**Despliega** " Error no se encuentran registrados los datos  
de la materia "

**Fin-Si**

**Sino**

**Despliega** "No se encuentra registrado el movimiento, o el  
movimiento no fue una baja o una licencia"

**Fin-Si**

MINIESPECIFICACIONES

6. Elabora Formato de Propuesta

**Define** variable v-num-movto, vcod-prog

**Define** Reporte Rep.Fmto-Propuesta

**/\* 6.1. Consulta Movimientos \*/**

**Obtiene** v-num-movto, vcod-prog

**Busca** num.movto en MOVIMIENTOS Donde

MOVIMIENTOS.num.movto = v-num-movto y

MOVIMIENTOS.cod-prog = vcod-prog

**Si** encuentra registro **Entonces Haz:**

**/\* 6.2. Consulta Profesor \*/**

**Busca** rfc en PROFESOR Donde PROFESOR.rfc =

MOVIMIENTOS.rfc

**Si** encuentra registro **Entonces Haz:**

**Envía** datos PROFESOR a Rep.Fmto-Propuesta

**/\* 6.3. Genera Fmto. Propuesta \*/**

**Imprime** Rep.Fmto-Propuesta

**Sino**

**Despliega** "Error en el movimiento,el profesor no tiene ese movimiento asignado"

**Fin-Si**

**Sino**

**Despliega** "El movimiento no se encuentra registrado"

**Fin-Si**

## CAPITULO III

### Diseño Estructurado.<sup>16</sup>

La siguiente etapa en el desarrollo de sistemas consiste en la transformación de los modelos generados en el análisis en los diagramas de estructura.

El diseño del sistema se inicia con el análisis del nivel 0. Primeramente, se especifica el nombre del sistema como el módulo principal o módulo ejecutivo, el cual al ser activado, ejecutara la tarea completa del sistema, a través de llamadas a sus módulos subordinados. Para cada uno de los procesos del nivel 0, se define un módulo subordinado al módulo principal.

La etapa del diseño continua con el análisis de los siguientes niveles de los diagramas de flujo de datos para identificar los flujos de transacción de los de transformación.

a) *Flujos de transacción:* Nos muestran los diferentes caminos que puede seguir el sistema, son módulos que a su vez son transformados en módulos ejecutivos que controlarán el acceso a otros caminos existentes; aunque también estos son módulos subordinados del módulo ejecutivo principal.

b) *Flujos de transformación:* Consisten en una secuencia de procesos que transforman las entradas del sistema en salidas. Cada flujo de transformación consiste en un módulo ejecutivo que ejecuta secuencialmente las operaciones de entrada (flujos aferentes), las operaciones de proceso (flujos de transformación) y las operaciones de salida (flujos eferentes).

El diseño de los flujos de transformación se lleva a cabo a su vez a través del análisis de transformación "El propósito de este análisis es identificar las

---

<sup>16</sup> Structured Design. Yourdon, Edward; Constantine, Larry. Ed. Prentice-Hall. USA, 1989. Capítulos 9 y 10.

funciones de procesamiento primarias del sistema, y las entradas y salidas de nivel más alto para esas funciones."<sup>17</sup>

El análisis de transformación se inicia con la identificación de los flujos aferentes y eferentes.

Para cada uno de los flujos de datos aferente, se crea un módulo subordinado inmediato del módulo ejecutivo anterior. Este módulo creado tendrá como función principal enviar los datos aferentes al módulo ejecutivo.

De manera similar, para cada uno de los elementos de datos aferentes, se define un módulo subordinado inmediato al módulo ejecutivo anterior, el cual aceptará los datos eferentes y los transformará en las salidas físicas finales.

Finalmente se especifica un módulo de transformación subordinado al módulo ejecutivo, el cual aceptará los datos de entrada y transformará estos datos en las salidas del sistema, las cuales serán enviadas de regreso al módulo ejecutivo.

El proceso anterior se conoce con el nombre de factorización y tiene como principal objetivo asegurar que los módulos de nivel más alto realicen funciones de control y coordinación, en tanto que los módulos de niveles más bajo, realizan funciones de procesamiento. "El módulo ejecutivo de nivel más alto no ejecuta ninguna de las tareas por sí solo, sin embargo si controla y coordina la ejecución de los módulos de nivel más bajo"

Del proceso de factorización se crea una estructura modular jerarquica inicial que debe ser refinada, basándose en los conceptos de cohesión y acoplamiento.

---

<sup>17</sup> Structured Design, Yourdon, Edward. Constantine, Larry. Ed. Prentice-Hall Hispanoamericana. U.S.A., 1989. p. 189.

Ambos terminos se definen a continuación:

**Cohesión:** "No fragmentar los procesos esenciales en módulos." <sup>18</sup>

**Acoplamiento:** "Grado en que los módulos se interconectan o se relacionan entre sí." <sup>19</sup>

A través del diseño estructurado determinamos cuales son las partes del sistema, qué se realiza en cada una, cómo están organizadas y cómo se relacionan entre sí.

La simbología utilizada en el diseño del sistema de Horarios de Profesores es la siguiente:

<b><i>Símbolo</i></b>	<b><i>Significado</i></b>
	<b>Módulo</b> Realiza una actividad específica del sistema
	<b>Línea</b> Muestra los diferentes caminos que puede tomar el sistema
	<b>Flujos de datos</b> Representa la entrada o salida de parámetros entre módulos Si la flecha apunta hacia arriba, significa un flujo de entrada Si la flecha apunta hacia abajo, significa un flujo de salida

---

<sup>18</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 465

<sup>19</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 465

## HORARIOS DE PROFESORES

### Diseño Estructurado

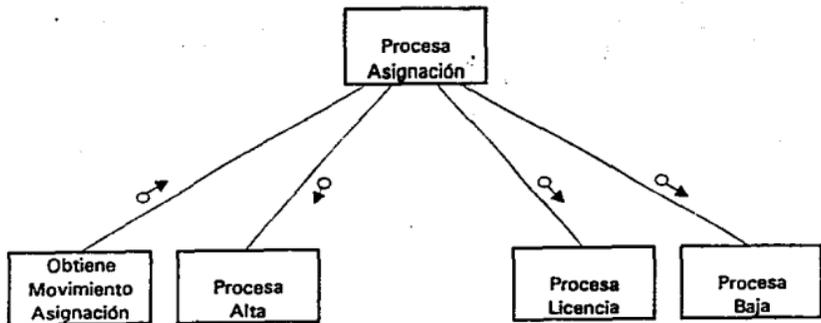
---

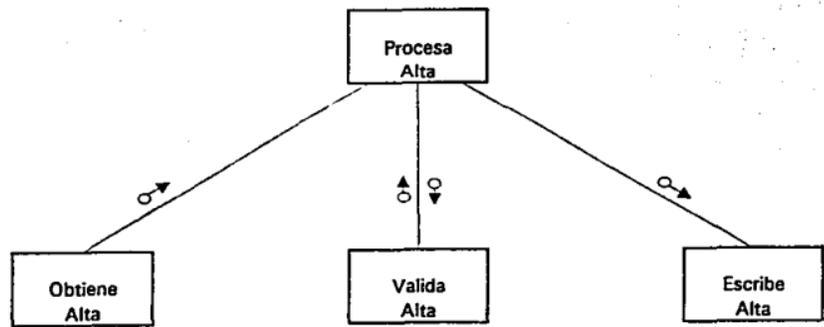
A continuación se presentan las convenciones de terminología que se utilizaron en el diseño del sistema de Horarios de Profesores:

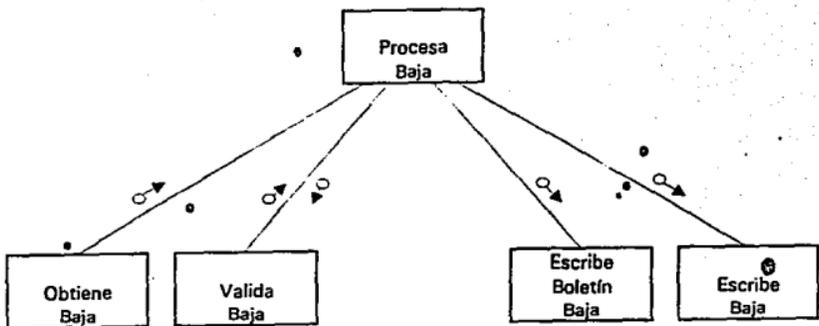
<b>Verbo</b>	<b>Significado</b>
Clasifica	El sistema evalúa datos proporcionados por el usuario y determina qué operaciones deben ejecutarse dependiendo de los resultados de dicha evaluación.
Elabora	El sistema toma información almacenada para construir un reporte de salida
Escribe	El sistema almacena información en archivos de datos
Genera	El sistema envía un reporte a impresora
Obtiene	El sistema toma información introducida por el usuario en pantalla
Procesa	El sistema aplica diversas operaciones a los datos de entrada para convertirlos en salidas
Valida	El sistema verifica que los datos proporcionados por el usuario cumplan con las políticas definidas en el sistema

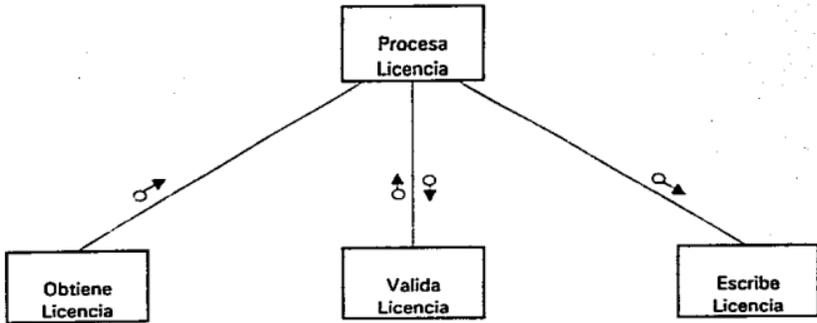
La definición de módulos del sistema de Horarios de Profesores se presenta en las siguientes páginas:

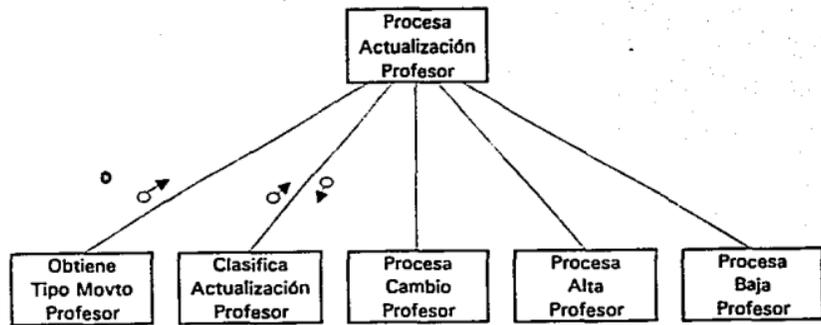


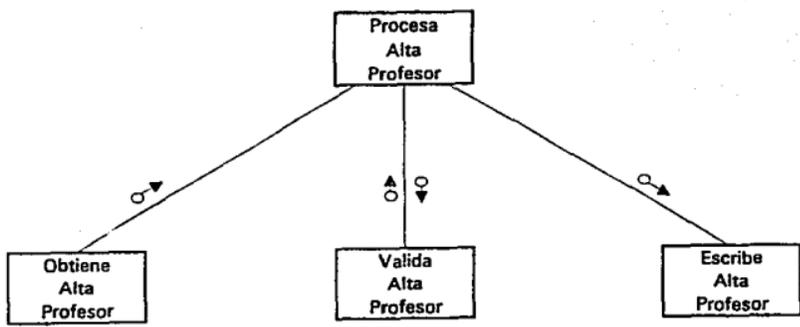


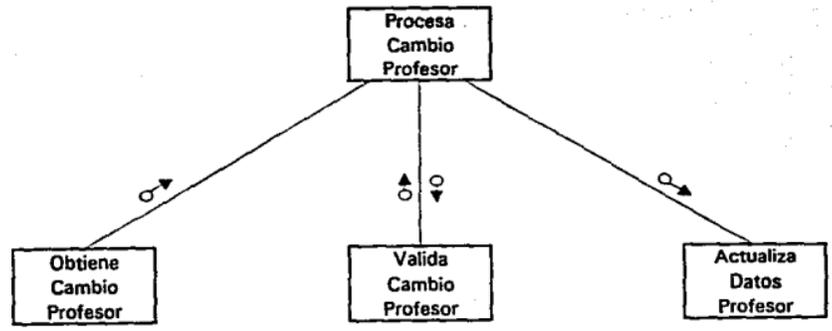


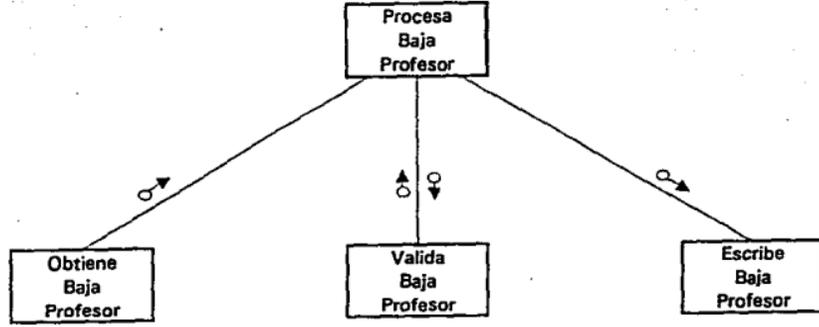


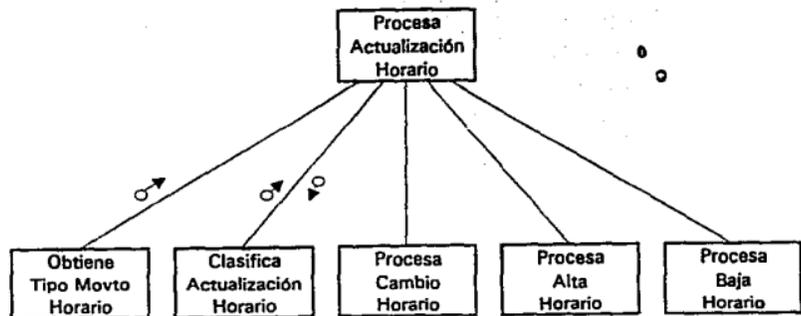


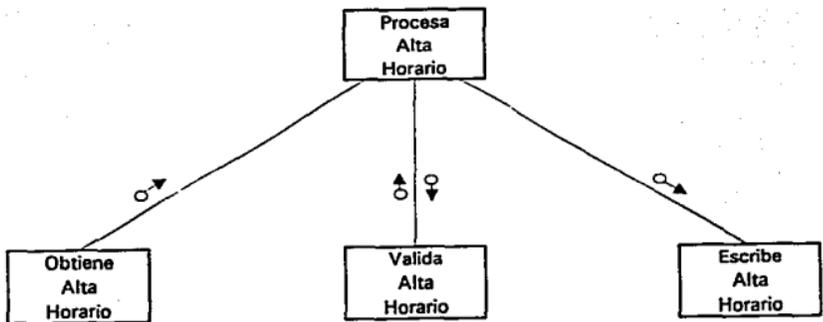


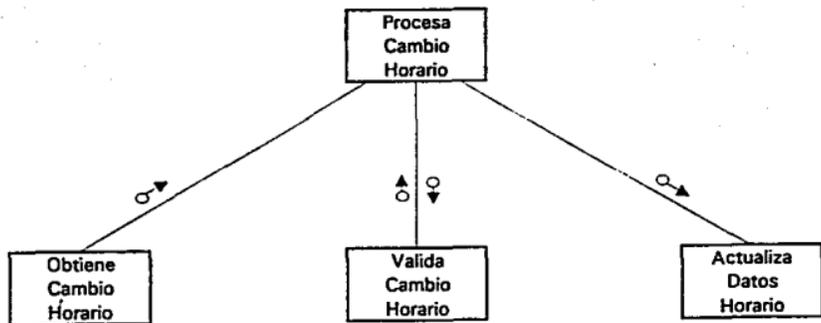


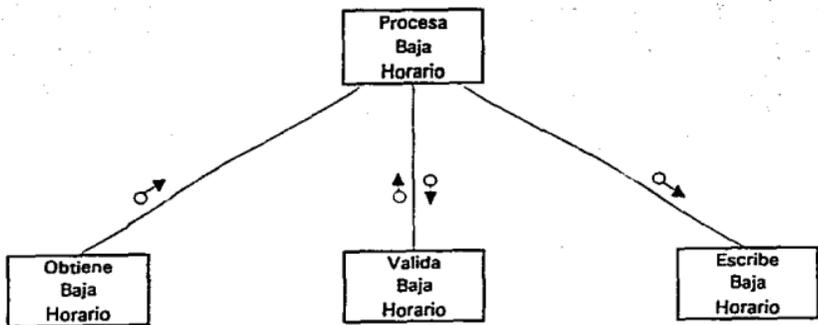


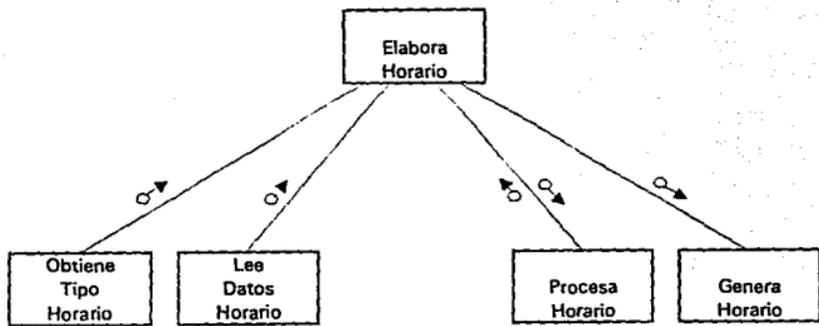


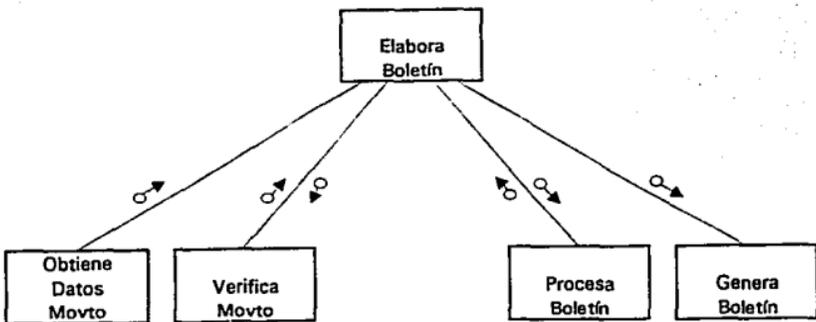


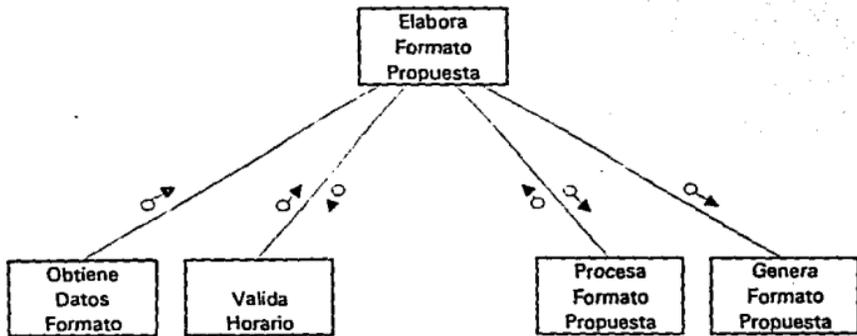












## CAPITULO IV

### Implementación

La fase de implementación consiste en "la codificación y la integración de módulos en un esqueleto progresivamente más completo del sistema final." <sup>20</sup>

La fase de programación o implementación implica la escritura de instrucciones en el lenguaje que se haya elegido para la construcción del sistema a partir de las especificaciones definidas en las fases de análisis y diseño del sistema. También involucra la ejecución de pruebas que ayuden a asegurarse de que el sistema produzca las salidas esperadas cumpliendo con los requerimientos que dieron origen a su creación.

En la programación del sistema de Horarios de Profesores se utilizó el lenguaje del manejador relacional de bases de datos Progress versión 6.0, que incluye diccionario de datos y facilidades para establecer ayudas en línea mediante la inclusión de mensajes en el diccionario de datos. Requiere 13 MB de espacio en disco duro, sistema operativo MSDOS 3.1 o superior y 640 KB en RAM.

La programación del sistema se realizó tomando como base la información disponible en la fase de diseño y en las miniespecificaciones y el diccionario de datos de la fase de análisis. De tal modo que para cada módulo presentado en los diagramas del diseño se elaboró un programa en Progress.

Esta etapa comenzó con la programación de los módulos de más alto nivel continuando hacia abajo en la estructura del sistema, los programas se fueron revisando con los usuarios para verificar que los menús y los formatos cubrieran con sus requerimientos.

Por otra parte, los nombres de los datos en los programas corresponden con los definidos en el diccionario de datos.

---

<sup>20</sup> Análisis Estructurado Moderno. Yourdon, Edward. Ed. Prentice-Hall Hispanoamericana. México, 1993. P. 102

Durante la programación es de vital importancia verificar que el sistema cubra con los requerimientos especificados en las fases de análisis y diseño. Para lograrlo se realizaron pruebas de funcionalidad partiendo desde los módulos de nivel más alto verificando que ejecutarán correctamente a los módulos subordinados. Posteriormente, se probaron los módulos de los siguientes niveles hasta llegar al último nivel del sistema.

"Este tipo de prueba conocido como Top-Down es una estrategia que prueba los módulos de más alto nivel, antes de que hayan sido codificados los módulos de bajo nivel y posiblemente antes de que hayan sido diseñados".<sup>21</sup>

La programación del sistema de Horarios de Profesores se presenta en el Anexo. Se incluyen los módulos principales del sistema.

---

<sup>21</sup> Managing Structured Techniques. Yourdon, Edward. Ed. Prentice-Hall. México, 1989. P. 67

## CONCLUSIONES

Las metodologías de análisis y diseño de sistemas representan un punto de partida para la creación de sistemas de información, al proporcionar herramientas para representar las necesidades de usuario y la forma en que el sistema habrá de cubrirías.

Las herramientas de la metodología nos permitieron establecer un eficiente canal de comunicación con el usuario, durante la etapa de análisis, siendo esta una etapa trascendente para el desarrollo del sistema al proporcionar los elementos básicos a partir de los cuales se estructurarán y definirán el objetivo, los límites y el alcance del mismo.

Es por tanto, que la metodología no debe ser limitativa, sino que el analista debe auxiliarse de todas aquellas herramientas que le permitan cubrir el objetivo y definir claramente el sistema.

No fue posible contar con una herramienta CASE que facilitara la creación y mantenimiento de las especificaciones del sistema, por lo que los modelos gráficos fueron elaborados con MS Draw, Excell y FlowChart.

En el desarrollo del Sistema de Horario de Profesores no fue necesario realizar el estudio de Factibilidad, como lo establece la metodología, debido a que la FES Zaragoza cuenta ya con sistemas automatizados desarrollados con herramientas de 4a. generación que operan en modo multiusuario y tiene interés en que los futuros sistemas se realicen bajo el mismo ambiente de trabajo, a fin de contar con un sistema integral.

Considerando este requerimiento, el Sistema de Horario de Profesores fue desarrollado utilizando el manejador de base de datos Progress V.6.0., bajo el sistema operativo Netware 3.11 de Novell.

Sin embargo, existe en la metodología una etapa de transición entre el análisis y el diseño en la que no se define con claridad la transformación de los modelos del análisis en diagramas de diseño estructurado y omite considerar los aspectos lógicos del sistema, por lo que realizamos la estructura del diseño considerando la lógica de los procesos definidos en los diagramas de flujo de datos, ya que a partir de éstos pudimos construir el modelo jerárquico funcional del sistema, al que posteriormente aplicamos los criterios de cohesión y acoplamiento para lograr un desempeño funcional más eficiente.

Es importante señalar que en la metodología de Yourdon el desarrollo de las actividades puede realizarse en forma paralela y sufrir modificaciones. En el desarrollo del sistema de Horario de Profesores algunos programas crecieron demasiado por lo que fue necesario redefinirlos, modificándose también las etapas de análisis y diseño, logrando así mantener la consistencia entre ellas.

En la metodología se presenta los archivos desde la etapa de análisis, sin mencionar en que momento fueron éstos definidos, por lo que los definimos conjuntamente con los diagramas de flujo de datos y las miniespecificaciones, realizando en esta misma etapa su normalización para evitar inconsistencia y redundancia.

El Sistema de Horarios de Profesores cumple con su objetivo original y aún puede seguir creciendo, ya que incluye en su estructura la información necesaria a partir de la cual se pueden generar de manera automática estadísticas que faciliten la toma de decisiones, tales como las relacionadas con el Banco de Horas, el mejor aprovechamiento de espacios al proporcionar información de la disponibilidad de salones, clínicas y laboratorios.

Las condiciones actuales del sistema no contemplan los criterios de depuración de los archivos por lo cual la información puede crecer y restar eficiencia al sistema.

Asimismo, carece de políticas para la creación de respaldos de la información, indispensables en caso de fallas o siniestros, que constituirían una herramienta de vital importancia para la reorganización de las actividades.

El sistema de Horarios de Profesores nos brindó la oportunidad de poner en práctica los conocimientos académicos adquiridos, enriqueciéndose con la experiencia de llevar a la práctica un proyecto, que además de que será útil a la Universidad puso a prueba nuestras habilidades como analistas y diseñadores de sistemas, sintiéndonos personalmente satisfechos de haber realizado nuestro trabajo.

## BIBLIOGRAFIA

### **Análisis Estructurado Moderno**

Yourdon, Edward

Ed. Prentice-Hall Hispanoamericana, S.A.

México, 1993

### **Structured Design**

Yourdon, Edward. Constantine, Larry

Prentice-Hall

USA, 1990

### **Progress Reference Manual**

Progress Software Corporation

USA

### **Progress User's Guide**

Progress Software Corporation

USA

## ANEXO

### Código Fuente

```
/* <MENUPRIN.P> menu principal */
```

```
DEF VAR J AS INTEGER.
```

```
DEF VAR FES AS CHAR FORMAT "X(13)" INITIAL "FES ZARAGOZA".
```

```
DEF VAR MENU1 AS CHAR EXTENT 6 FORMAT "X(26)"
```

```
  INITIAL ["1. PROCESA ASIGNACION",
```

```
    "2. ACTUALIZA PROFESOR",
```

```
    "3. ACTUALIZA HORARIO",
```

```
    "4. PROCESA HORARIO",
```

```
    "5. ELABORA BOLETIN",
```

```
    "6. ELABORA FMTO. PROPUESTA"].
```

```
DEF VAR programa AS CHAR EXTENT 6 FORMAT "X(11)"
```

```
  INITIAL ["MENUASIG.P",
```

```
    "MENUPROF.P",
```

```
    "MENUHORA.P",
```

```
    "HORARIO.P",
```

```
    "BOLETIN.P",
```

```
    "FORMPROP.P"].
```

```
FORM HEADER
```

```
FES AT 2 "MENU PRINCIPAL" AT 33 TODAY AT 69
```

```
FORMAT "99-99-9999" WITH PAGE-TOP WIDTH 80 TITLE "S C O H P"
```

```
CENTER FRAME TITULO ROW 1.
```

REPEAT:

HIDE ALL.

VIEW FRAME TITULO.

PUT SCREEN ROW 5 COLUMN 37 COLOR MESSAGE " MENU ".

DISPLAY MENU1 WITH NO-LABEL FRAME MEN1 1 COLUMN CENTER ROW 6.  
MESSAGE

" \*" + CHR(17) + CHR(196) + CHR(196) + CHR(217) + "\*" SELECCIONAR OPCION  
\*" + CHR(24) + "\*" ANTERIOR "\*" + CHR(25) "\*" SIGUIENTE "F4" SALIR ".

CHOOSE FIELD MENU1 AUTO-RETURN WITH FRAME MEN1.

IF CAN-DO ("RETURN,PICK,GO",KEYFUNCTION(LASTKEY)) THEN

DO J= 1 TO 6:

IF FRAME-VALUE = MENU1[J] THEN LEAVE.

END.

ELSE DO:

BELL.

NEXT.

END.

HIDE MESSAGE.

PAUSE 0.

HIDE ALL.

IF SEARCH(PROGRAMA[J]) = ? THEN

MESSAGE "EL PROGRAMA:" PROGRAMA[J] "NO EXISTE".

ELSE RUN VALUE(PROGRAMA[J]).

END.

/\*QUIT\*/

/\* <MENUASIG.P> menu de asignacion de profesores \*/

```
DEF VAR J AS INTEGER.
DEF VAR FES AS CHAR FORMAT "X(13)" INITIAL "FES ZARAGOZA".
DEF VAR MENU1 AS CHAR EXTENT 3 FORMAT "X(18)"
  INITIAL ["1. ASIGNA ALTA",
    "2. ASIGNA LICENCIA",
    "3. ASIGNA BAJA"].
DEF VAR PROGRAMA AS CHAR EXTENT 8 FORMAT "X(10)"
  INITIAL ["ASIGALTA.P","ASIGLIC.P","ASIGBAJA.P"].
```

FORM HEADER

```
FES AT 2 "MENU DE ASIGNACIONES" AT 30 TODAY AT 69
FORMAT "99-99-9999" WITH PAGE-TOP WIDTH 80 TITLE "S C O H P"
CENTER FRAME TITULO ROW 1.
```

REPEAT:

```
HIDE ALL.
VIEW FRAME TITULO.
PUT SCREEN ROW 5 COLUMN 37 COLOR MESSAGE " MENU ".
DISPLAY MENU1 WITH NO-LABEL FRAME MEN1 1 COLUMN CENTER ROW 6.
MESSAGE
" * + CHR(17) + CHR(198) + CHR(196) + CHR(217) + "SELECCIONAR OPCION
* + CHR(24) + " ANTERIOR * + CHR(25) + " SIGUIENTE *F4* SALIR ".
CHOOSE FIELD MENU1 AUTO-RETURN WITH FRAME MEN1.
IF CAN-DO("RETURN,PICK,GO",KEYFUNCTION(LASTKEY)) THEN
DO J= 1 TO 3:
  IF FRAME-VALUE = MENU1[J] THEN LEAVE.
END.
ELSE DO:
  BELL.
  NEXT.
END.
HIDE MESSAGE.
PAUSE 0.
HIDE ALL.
IF SEARCH (PROGRAMA[J]) = ? THEN
  MESSAGE "EL PROGRAMA: " PROGRAMA[J] "NO EXISTE".
ELSE RUN VALUE (PROGRAMA[J]).
END.
/*QUIT*/
```

/\* <MENU PROF.P> menu de actualizacion de profesores \*/

```
DEF VAR J AS INTEGER.
DEF VAR FES AS CHAR FORMAT "X(13)" INITIAL "FES ZARAGOZA".
DEF VAR MENU1 AS CHAR EXTENT 3 FORMAT "X(18)"
  INITIAL ["1. ALTA PROFESOR",
    "2. BAJA PROFESOR",
    "3. CAMBIA PROFESOR"].
DEF VAR PROGRAMA AS CHAR EXTENT 8 FORMAT "X(10)"
  INITIAL ["ALTAPROF.P", "BAJAPROF.P", "CAMPROF.P"].
```

FORM HEADER

```
FES AT 2 "MENU DE PROFESORES" AT 30 TODAY AT 69
FORMAT "99-99-9999" WITH PAGE-TOP WIDTH 80 TITLE "S C O H P"
CENTER FRAME TITULO ROW 1.
```

REPEAT:

```
HIDE ALL.
VIEW FRAME TITULO.
PUT SCREEN ROW 5 COLUMN 37 COLOR MESSAGE " MENU ".
DISPLAY MENU1 WITH NO-LABEL FRAME MEN1 1 COLUMN CENTER ROW 6.
MESSAGE
" * + CHR(17) + CHR(196) + CHR(196) + CHR(217) + "SELECCIONAR OPCION
" + CHR(24) + " ANTERIOR  " + CHR(25) + " SIGUIENTE  " + CHR(24) + " SALIR ".
CHOOSE FIELD MENU1 AUTO-RETURN WITH FRAME MEN1.
IF CAN-DO("RETURN,PICK,GO",KEYFUNCTION(LASTKEY)) THEN
  DO J= 1 TO 3:
    IF FRAME-VALUE = MENU1[J] THEN LEAVE.
  END.
ELSE DO:
  BELL.
  NEXT.
END.
HIDE MESSAGE.
PAUSE 0.
HIDE ALL.
IF SEARCH (PROGRAMA[J]) = ? THEN
  MESSAGE "EL PROGRAMA: " PROGRAMA[J] "NO EXISTE".
  ELSE RUN VALUE (PROGRAMA[J]).
END.
/*QUIT*/
```

/\* <MENUHORA.P> menu de actualizacion de horarios \*/

DEF VAR J AS INTEGER.

DEF VAR FES AS CHAR FORMAT "X(13)" INITIAL "FES ZARAGOZA".

DEF VAR MENU1 AS CHAR EXTENT 3 FORMAT "X(19)"

INITIAL ["1. REGISTRA HORARIO",

"2. BORRA HORARIO",

"3. CAMBIA HORARIO"].

DEF VAR PROGRAMA AS CHAR EXTENT 8 FORMAT "X(10)"

INITIAL ["REGHORA.P","BORRAHOR.P","CAMBHORA.P"].

FORM HEADER

FES AT 2 "MENU DE HORARIOS" AT 30 TODAY AT 69

FORMAT "99-99-9999" WITH PAGE-TOP WIDTH 80 TITLE "S C O H P"

CENTER FRAME TITULO ROW 1.

REPEAT:

HIDE ALL.

VIEW FRAME TITULO.

PUT SCREEN ROW 5 COLUMN 37 COLOR MESSAGE " MENU ".

DISPLAY MENU1 WITH NO-LABEL FRAME MEN1 1 COLUMN CENTER ROW 6.

MESSAGE " \*" + CHR(17) + CHR(196) + CHR(196) + CHR(217) +

"SELECCIONAR OPCION \*" + CHR(24) + "\*" ANTERIOR \*" + CHR(25) + "\*" SIGUIENTE \*F4\* SALIR ".

CHOOSE FIELD MENU1 AUTO-RETURN WITH FRAME MEN1.

IF CAN-DO("RETURN,PICK,GO",KEYFUNCTION(LASTKEY)) THEN

DO J= 1 TO 3:

IF FRAME-VALUE = MENU1[J] THEN LEAVE.

END.

ELSE DO:

BELL.

NEXT.

END.

HIDE MESSAGE.

PAUSE 0.

HIDE ALL.

IF SEARCH (PROGRAMA[J]) = ? THEN

MESSAGE "EL PROGRAMA: " PROGRAMA[J] "NO EXISTE".

ELSE RUN VALUE (PROGRAMA[J]).

END.

/\*QUIT\*/

/\* BOLETIN.P PROGRAMA PARA LA ELABORACION DEL BOLETIN DE HORAS VACANTES QUE SE GENERA CUANDO HAY UNA BAJA O ALTA DE PROFESOR \*/

```

DEF NEW SHARED VAR VNO-MOVTO LIKE MOVTO.NO-MOVTO.
DEF NEW SHARED VAR VNO-BOLETIN AS CHA FORMAT "99999999-99".
DEF NEW SHARED VAR VCARRERA AS CHA FORMAT "XXXXXXXXXXXXXXXXXXXX".
DEF NEW SHARED VAR VUNIDAD AS CHA FORMAT "XXXXXXXXXXXXXXXX".
DEF NEW SHARED VAR VANIO AS CHA FORMAT "XXXXXX".
DEF NEW SHARED VAR VCOD-PROG AS CHA FORMAT "9999".
DEF NEW SHARED VAR VTIPO-MOVTO LIKE MOVTO.TIPO-MOVTO.
DEF NEW SHARED VAR VFUNCION AS CHA FORMAT "XXXXXXXXXX".
DEF NEW SHARED VAR VCVE-CAUSA AS CHA FORMAT
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX".
DEF VAR EXISTE AS LOG.
DEF VAR RESP AS CHA FORMAT "X".
DEF VAR VATEGORIA AS CHA FORMAT "XXXXXXXXXXXXXXXXXXXX".
DEF VAR VSUMHORA AS INTEGER FORMAT 99.
DEF VAR VSUMHORA1 AS INTEGER FORMAT 99.
DEF NEW SHARED VAR VCICLO LIKE HORARIO.CICLO.
DEF NEW SHARED VAR VASIGNATURA LIKE HORARIO.ASIGNATURA.
DEF NEW SHARED VAR VACTIVIDAD LIKE HORARIO.ACTIVI.
DEF NEW SHARED VAR VGRUPO LIKE HORARIO.GRUPO.
DEF NEW SHARED VAR VDIA-1 LIKE HORARIO.DIA-1.
DEF NEW SHARED VAR VHOR-INI1 LIKE HORARIO.HOR-INI1.
DEF NEW SHARED VAR VHOR-FIN1 LIKE HORARIO.HOR-FIN1.
DEF NEW SHARED VAR VDIA-2 LIKE HORARIO.DIA-2.
DEF NEW SHARED VAR VHOR-INI2 LIKE HORARIO.HOR-INI2.
DEF NEW SHARED VAR VHOR-FIN2 LIKE HORARIO.HOR-FIN2.
DEF NEW SHARED VAR VVIGENCIA LIKE MOVTO.VIGENCIA.
DEF NEW SHARED VAR VFUNCION1 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VFUNCION2 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VFUNCION3 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VFUNCION4 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VFUNCION5 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VFUNCION6 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VREQUISITO1 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VREQUISITO2 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VREQUISITO3 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VREQUISITO4 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VREQUISITO5 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VREQUISITO6 AS CHA FORMAT "X(60)".
DEF NEW SHARED VAR VLUGAR AS CHA FORMAT "X(20)".
    
```

VCATEGORIA = " ".  
RESP = "N".  
VCARRERA = "CIRUJANO DENTISTA".  
VUNIDAD = "FES ZARAGOZA".

FORM " " SKIP

"UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO " AT 20 SKIP  
"CONTROL DE HORARIOS DE PROFESORES FESZ ZARAGOZA " AT 16 SKIP  
WITH WIDTH 80 CENTERED FRAME TITULO.

FORM

SKIP(2)

VNO-MOVTO LABEL "NO DE MOVIMIENTO" AT 25 SPACE(15) SKIP(1)

VCOD-PROG LABEL " COD PROG"

VCARRERA LABEL " CARRERA " SKIP (1)

HORARIO.ASIGNATURA LABEL " MATERIA " SKIP (1)

HORARIO.DIA-1 LABEL " HORARIO " "DE" HORARIO.HOR-INI1 "A"

HORARIO.HOR-FIN1

"Y" HORARIO.DIA-2 "DE" HORARIO.HOR-INI2 "A" HORARIO.HOR-FIN2

SKIP(1)

VFUNCION LABEL " CARGO " SKIP (1)

VLUGAR LABEL " LUGAR" VNO-BOLETIN LABEL "NO DE BOLETIN"

SKIP(1)

WITH WIDTH 70 CENTERED SIDE-LABELS

TITLE "ELABORACION DE BOLETIN DE HORAS VACANTES" FRAME

FORMATO.

FORM

SKIP(2)

VREQUISITO1 LABEL "REQUISITO 1" SKIP

VREQUISITO2 LABEL "REQUISITO 2" SKIP

VREQUISITO3 LABEL "REQUISITO 3" SKIP

VREQUISITO4 LABEL "REQUISITO 4" SKIP

VREQUISITO5 LABEL "REQUISITO 5" SKIP

VREQUISITO6 LABEL "REQUISITO 6" SKIP

VFUNCION1 LABEL "FUNCION 1" SKIP

VFUNCION2 LABEL "FUNCION 2" SKIP

VFUNCION3 LABEL "FUNCION 3" SKIP

VFUNCION4 LABEL "FUNCION 4" SKIP

VFUNCION5 LABEL "FUNCION 5" SKIP

VFUNCION6 LABEL "FUNCION 6" SKIP  
 WITH WIDTH 75 CENTERED SIDE-LABELS  
 TITLE "REQUISITOS Y FUNCIONES PARA BOLETIN DE HORAS VACANTES"  
 FRAME FORMATO2.

FORM HEADER

"ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES ZARAGOZA" AT 15 SKIP  
 "CARRERA DE CIRUJANO DENTISTA" AT 27 SKIP  
 "BOLETIN No." AT 29 VNO-BOLETIN SKIP(2)  
 "De acuerdo al procedimiento de SUSTITUCION DE PROFESORES. Se hace del  
 conoci-"  
 SKIP  
 "miento que se requiere cubrir las actividades docentes que corresponden a la"  
 SKIP  
 "siguiente especificacion:" SKIP(1)  
 "CARRERA : " VCARRERA  
 "  
 \_\_\_\_\_"  
 SKIP  
 "CICLO : " VCICLO  
 "  
 \_\_\_\_\_"  
 " SKIP  
 "MATERIA O MODULO : " VASIGNATURA " \_\_\_\_\_"  
 SKIP  
 "MODALIDAD : " VSUMHORA "HRS." VACTIVIDAD  
 "  
 \_\_\_\_\_"  
 " SKIP  
 "GRUPO : " VGRUPO  
 "  
 \_\_\_\_\_"  
 " SKIP  
 "No. DE HORAS : " VSUMHORA1 "HRS."  
 "  
 \_\_\_\_\_"  
 " SKIP  
 "HORARIO : " VDIA-1 "DE" VHOR-INI1 "a" VHOR-FIN1 "y" VDIA-2 "DE" VHOR-INI2 "a"  
 VHOR-FIN2 " \_\_\_\_\_" SKIP  
 "LUGAR : " VLUGAR " \_\_\_\_\_"  
 SKIP  
 "CATEGORIA : " VCATEGORIA  
 "  
 \_\_\_\_\_"  
 SKIP  
 "CARGO : " VFUNCION  
 "  
 \_\_\_\_\_"  
 " SKIP

"CAUSA DE LA SUSTITUCION : " VCVE-CAUSA " \_\_\_\_\_ "  
 SKIP

"PERIODO : " A PARTIR DE " VVIGENCIA

" \_\_\_\_\_ "  
 " SKIP

WITH WIDTH 80 CENTERED FRAME EN CAB.

FORM

SKIP

"REQUISITOS " SKIP(1)

VREQUISITO1 SKIP

VREQUISITO2 SKIP

VREQUISITO3 SKIP

VREQUISITO4 SKIP

VREQUISITO5 SKIP

VREQUISITO6 SKIP

VREQUISITO6 SKIP(1)

"FUNCION " SKIP(1)

VFUNCION1 SKIP

VFUNCION2 SKIP

VFUNCION3 SKIP

VFUNCION4 SKIP

VFUNCION5 SKIP

VFUNCION6 SKIP(1)

"Los interesados podran recoger su solicitud en la carrera del Departamento"

SKIP

"de Asuntos de Personal Academico. La cual debera entregarla acompañada de los"

SKIP

"documentos probatorios correspondientes dos juegos, a partir de esta fecha"

SKIP

"hasta a las 18:00 horas."

SKIP (3)

"A T E N T A M E N T E" SKIP

"POR MI RAZA HABLARA EL ESPIRITU" SKIP

"MEXICO, D.F. A " SKIP

"EL JEFE DE LA CARRERA" SKIP

WITH WIDTH 80 CENTERED NO-LABELS FRAME DETALLE.

REPEAT:

HIDE ALL.

---

```

EXISTE = FALSE.
HIDE MESSAGE.
MESSAGE "F4 SALIR".
VIEW FRAME TITULO.
clear frame formato all.
UPDATE VNO-MOVTO help "NO. MOVIMIENTO DE LA ASIGNACION"
VCOD-PROG help "CODIGO PROGRAMATICO DE LA ASIGNATURA"
WITH FRAME FORMATO.
FIND MOVTO WHERE MOVTO.NO-MOVTO = VNO-MOVTO AND MOVTO.COD-
PROG = VCOD-PROG
NO-ERROR.
IF NOT AVAILABLE MOVTO THEN DO:
MESSAGE "EL MOVIMIENTO NO SE ENCUENTRA REGISTRADO O EL
MOVTO NO ".
MESSAGE "FUE UNA LICENCIA O UNA BAJA POR LO QUE NO SE GENERA
BOL".
PAUSE NO-MESSAGE.
VNO-MOVTO = " ";
VCOD-PROG = " ";
NEXT.
END.
ELSE DO:
FIND HORARIO WHERE HORARIO.COD-PROG = VCOD-PROG AND
HORARIO.CVE-FUN = MOVTO.CVE-FUN NO-ERROR.
IF NOT AVAILABLE HORARIO THEN DO:
MESSAGE "ERROR EN EL MOVIMIENTO".
PAUSE NO-MESSAGE.
VNO-MOVTO = " ";
VCOD-PROG = " ";
NEXT.
END.
ELSE DO:
IF HORARIO.CVE-FUN = "1" OR HORARIO.CVE-FUN = "2" THEN
VFUNCION = "TITULAR".
IF HORARIO.CVE-FUNC = "3" OR HORARIO.CVE-FUNC = "4" OR
HORARIO.CVE-FUNC = "5" THEN VFUNCION = "AYUDANTE".
DISPLAY VNO-MOVTO VCOD-PROG VCARRERA
HORARIO.ASIGNATURA HORARIO.DIA-1 HORARIO.HOR-INI1
HORARIO.HOR-FIN1
HORARIO.DIA-2 HORARIO.HOR-INI2 HORARIO.HOR-FIN2 VFUNCION
WITH FRAME FORMATO.

UPDATE VLUGAR help "LUGAR DONDE SE IMPARTIRA LA ASIGNATURA"

```

---

VNO-BOLETIN help "NO DE BOLETIN"  
 WITH FRAME FORMATO.  
 PAUSE NO-MESSAGE.  
 DISPLAY " QUIERES IMPRIMIR EL BOLETIN DE HORAS VACANTES"  
 WITH FRAME FORMATO.  
 UPDATE RESP help "TECLEA S/N" WITH NO-LABEL FRAME FORMATO.  
 IF RESP = "S" THEN DO:  
 UPDATE VREQUISITO1 help "TECLEE LOS REQUISITOS PARA EL  
 BOLETIN"  
 VREQUISITO2 VREQUISITO3 VREQUISITO4 VREQUISITO5  
 VREQUISITO6  
 VFUNCION1 help "TECLEE LAS FUNCIONES PARA EL BOLETIN"  
 VFUNCION2 VFUNCION3 VFUNCION4 VFUNCION5 VFUNCION6  
 WITH FRAME FORMATO2.  
 IF MOVTO.CVE-CAUSA = "21" THEN VCVE-CAUSA = "OTRO  
 NOMBRAMIENTO".  
 IF MOVTO.CVE-CAUSA = "22" THEN VCVE-CAUSA =  
 "RECLASIFICACION".  
 IF MOVTO.CVE-CAUSA = "23" THEN VCVE-CAUSA = "TRANSFERENCIA".  
 IF MOVTO.CVE-CAUSA = "24" THEN VCVE-CAUSA = "DISMINUCION  
 HORAS".  
 IF MOVTO.CVE-CAUSA = "25" THEN VCVE-CAUSA = "ENFERMEDAD".  
 IF MOVTO.CVE-CAUSA = "26" THEN VCVE-CAUSA = "GRAVIDEZ".  
 IF MOVTO.CVE-CAUSA = "27" THEN VCVE-CAUSA = "SUSPENSION".  
 IF MOVTO.CVE-CAUSA = "28" THEN VCVE-CAUSA = "RENUNCIA".  
 IF MOVTO.CVE-CAUSA = "29" THEN VCVE-CAUSA = "RESCISION".  
 IF MOVTO.CVE-CAUSA = "30" THEN VCVE-CAUSA = "DEFUNCION".  
 IF MOVTO.CVE-CAUSA = "31" THEN VCVE-CAUSA = "A#O SABATICO".  
 IF MOVTO.CVE-CAUSA = "32" THEN VCVE-CAUSA = "COMISION  
 OFICIAL".  
 IF MOVTO.CVE-CAUSA = "33" THEN VCVE-CAUSA = "ESTUDIOS EN EL  
 PAIS".  
 IF MOVTO.CVE-CAUSA = "34" THEN VCVE-CAUSA = "ESTUDIOS EN EL  
 EXTR".  
 IF MOVTO.CVE-CAUSA = "35" THEN VCVE-CAUSA = "PARTICULAR".  
 IF HORARIO.CVE-CATEG = "1A" THEN VCATEGORIA = "AYUDANTE A".  
 IF HORARIO.CVE-CATEG = "1B" THEN VCATEGORIA = "AYUDANTE B".  
 IF HORARIO.CVE-CATEG = "2A" THEN VCATEGORIA = "ASIGNATURA A".  
 IF HORARIO.CVE-CATEG = "2B" THEN VCATEGORIA = "ASIGNATURA B".  
 IF HORARIO.CVE-CATEG = "3A" THEN VCATEGORIA = "ASIG A  
 DEFINITIVO".  
 IF HORARIO.CVE-CATEG = "3B" THEN VCATEGORIA = "ASIG B  
 DEFINITIVO".  
 OUTPUT TO PRINTER PAGE-SIZE 80.

---

VSUMHORA = HORARIO.HOR-FIN1 - HORARIO.HOR-INI1 +  
HORARIO.HOR-FIN2 - HORARIO.HOR-INI2.

VSUMHORA1 = VSUMHORA.

VCICLO = HORARIO.CICLO.  
VASIGNATURA = HORARIO.ASIGNATURA.  
VACTIVIDAD = HORARIO.ACTIVI.  
VGRUPO = HORARIO.GRUPO.  
VDIA-1 = HORARIO.DIA-1.  
VHOR-INI1 = HORARIO.HOR-INI1.

VHOR-FIN1 = HORARIO.HOR-FIN1.  
VDIA-2 = HORARIO.DIA-2.  
VHOR-INI2 = HORARIO.HOR-INI2.  
VHOR-FIN2 = HORARIO.HOR-FIN2.  
VIGENCIA = MOVTO.VIGENCIA.

VIEW FRAME EN CAB.

DISPLAY VREQUISITO1 VREQUISITO2 VREQUISITO3 VREQUISITO4  
VREQUISITO5 VREQUISITO6 VFUNCION1 VFUNCION2 VFUNCION3  
VFUNCION4 VFUNCION5 VFUNCION6

WITH FRAME DETALLE.

END.

END.

END.

OUTPUT CLOSE.

VNO-MOVTO = " ".

VCOD-PROG = " ".

END.

/\* FORMPROP.P\*/

DEF NEW SHARED VAR VNO-MOVTO LIKE MOVTO.NO-MOVTO.  
DEF NEW SHARED VAR VNOMBRE LIKE PROFESOR.NOMBRE.  
DEF NEW SHARED VAR VCICLO AS CHA FORMAT "XXXX".  
DEF NEW SHARED VAR VCARRERA AS CHA FORMAT "XXXXXXXXXXXXX".  
DEF NEW SHARED VAR VUNIDAD AS CHA FORMAT "XXXXXXXXX".  
DEF NEW SHARED VAR VALTA AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VBAJA AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VLIC AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VCSDO AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VSSDO AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VREMADI AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VRECLAS AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VANIOS AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VDEFUN AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VNVOINGR AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VAUMHOR AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VTRANSF AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VCOMOFIC AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VESTPAIS AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VREINGR AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VPROM AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VDISHOR AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VREVOC AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VESTEXT AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VPRORRO AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VREANUD AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VENFERM AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VSUSPEN AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VPARTIC AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VOTRON AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VINTER AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VGRAVI AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VRENUN AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VRESIC AS CHA FORMAT "XX".  
DEF NEW SHARED VAR VPROF LIKE PROFESOR.NOMBRE.  
DEF NEW SHARED VAR VRFC AS CHA FORMAT "XXXXXXXXXXXXX".  
DEF NEW SHARED VAR VCOD-PROG AS CHA FORMAT "9999".  
DEF NEW SHARED VAR VTIPO-MOVTO LIKE MOVTO.TIPO-MOVTO.  
DEF NEW SHARED VAR VCVE-CAUSA LIKE MOVTO.CVE-CAUSA.  
DEF NEW SHARED VAR VGRUPO AS CHA FORMAT "9999".  
DEF VAR EXISTE AS LOG.

DEF VAR RESP AS CHA FORMAT "X".  
 DEF VAR VLU1 AS INTEGER FORMAT 99.  
 DEF VAR VLU2 AS INTEGER FORMAT 99.  
 DEF VAR VMA1 AS INTEGER FORMAT 99.  
 DEF VAR VMA2 AS INTEGER FORMAT 99.  
 DEF VAR VMI1 AS INTEGER FORMAT 99.  
 DEF VAR VMI2 AS INTEGER FORMAT 99.  
 DEF VAR VJU1 AS INTEGER FORMAT 99.  
 DEF VAR VJU2 AS INTEGER FORMAT 99.  
 DEF VAR VV1 AS INTEGER FORMAT 99.  
 DEF VAR VV2 AS INTEGER FORMAT 99.  
 DEF VAR VCATEGORIA AS CHA FORMAT "XXXXXXXXXXXXXXXXXXXX".  
 DEF VAR VTEORIA AS INTEGER FORMAT 99.  
 DEF VAR VPRACTICA AS INTEGER FORMAT 99.  
 DEF VAR SUMTEO AS INTEGER FORMAT 99.  
 DEF VAR SUPRA AS INTEGER FORMAT 99.

VCATEGORIA = " ".  
 VTEORIA = 0.  
 VPRACTICA = 0.

RESP = "N".

VCARRERA = "ODONTOLOGIA".  
 VCICLO = "94-1".  
 VUNIDAD = "ZARAGOZA".  
 VALTA = " \_ ".  
 VBAJA = " \_ ".  
 VLIC = " \_ ".  
 VCSDO = " \_ ".  
 VSSDO = " \_ ".  
 VREMADI = " \_ ".  
 VRECLAS = " \_ ".  
 VANIOS = " \_ ".  
 VDEFUN = " \_ ".  
 VNVOINGR = " \_ ".  
 VAUMHOR = " \_ ".  
 VTRANSF = " \_ ".  
 VCOMOFIC = " \_ ".  
 VESTPAIS = " \_ ".  
 VREINGR = " \_ ".  
 VPROM = " \_ ".  
 VDISHOR = " \_ ".  
 VREVOC = " \_ ".

VESTEXT = " \_."  
VPRORRO = " \_."  
VREANUD = " \_."  
VENFERM = " \_."  
VSUSPEN = " \_."  
VPARTIC = " \_."  
VOTRON = " \_."  
VINTER = " \_."  
VGRAVI = " \_."  
VRENUN = " \_."  
VRESIC = " \_."

FORM " " SKIP

"UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO " AT 20 SKIP  
"CONTROL DE HORARIOS DE PROFESORES FESZ ZARAGOZA " AT 16 SKIP  
WITH WIDTH 80 CENTERED FRAME TITULO.

FORM

SKIP(2)  
VNO-MOVTO LABEL "NO DE MOVIMIENTO" AT 25 SPACE(15) SKIP(1)  
VCOD-PROG LABEL " COD PROG"  
PROFESOR.NOMBRE LABEL " PROFESOR" SKIP(1)  
MOVTO.RFC LABEL " RFC"  
MOVTO.TIPO-MOVTO LABEL " TIPO MOVTO"  
MOVTO.CVE-CAUSA LABEL "CAUSA" SKIP(2)  
WITH WIDTH 70 CENTERED ROW 8 SIDE-LABELS  
TITLE "ELABORACION DE FORMATO DE PROPUESTA" FRAME FORMATO.

FORM HEADER

"ESCUELA NACIONAL DE ESTUDIOS SUPERIORES ZARAGOZA      CICLO:"  
VCICLO SKIP  
SKIP (1)  
"CARRERA / COORDINACION / UNIDAD:" VCARRERA VUNIDAD SKIP(1)  
VALTA "ALTA " VBAJA AT 30 "BAJA " VLIC AT 54 "LICENCIA" VCSDO AT 80  
"CON SUELDO"  
VSSDO AT 103 "SIN SUELDO" SKIP  
SKIP (1)  
"POR CAUSA DE : " VREMADI AT 30 "REMUN.ADICIONAL" VRECLAS AT 54  
"RECLASIFICACION"  
VANIOS AT 80  
"AÑO SABATICO" VDEFUN AT 103 "DEFUNCION" SKIP



---

WITH WIDTH 132 CENTERED NO-LABELS FRAME DETALLE.

```

REPEAT:
  HIDE ALL.
  EXISTE = FALSE.
  HIDE MESSAGE.
  MESSAGE "F4 SALIR".
  VIEW FRAME TITULO.
  clear frame formato all.
  UPDATE VNO-MOVTO help "NO. MOVIMIENTO DE LA ASIGNACION"
    VCOD-PROG help "CODIGO PROGRAMATICO DE LA ASIGNATURA"
  WITH FRAME FORMATO.
  FIND MOVTO WHERE MOVTO.NO-MOVTO = VNO-MOVTO AND MOVTO.COD-
PROG =
    VCOD-PROG NO-ERROR.
  IF NOT AVAILABLE MOVTO THEN DO:
    MESSAGE "EL MOVIMIENTO NO SE ENCUENTRA REGISTRADO".
    PAUSE NO-MESSAGE.
    VNO-MOVTO = " ".
    VCOD-PROG = " ".
    NEXT.
  END.
  ELSE DO:
    FIND PROFESOR WHERE MOVTO.RFC = PROFESOR.RFC NO-ERROR.
    IF NOT AVAILABLE PROFESOR THEN DO:
      MESSAGE "ERROR EN EL MOVIMIENTO".
      PAUSE NO-MESSAGE.
      VNO-MOVTO = " ".
      VCOD-PROG = " ".
      NEXT.
    END.
  ELSE DO:
    DISPLAY VNO-MOVTO VCOD-PROG PROFESOR.NOMBRE MOVTO.RFC
    MOVTO.TIPO-MOVTO MOVTO.CVE-CAUSA WITH FRAME FORMATO.
    PAUSE NO-MESSAGE.
    DISPLAY "          QUIERES IMPRIMIR EL FORMATO DE PROPUESTA"
    WITH FRAME FORMATO.

    UPDATE RESP help "TECLEA S/N" WITH NO-LABEL FRAME FORMATO.
    IF RESP = "S" THEN DO:
      VNOMBRE = PROFESOR.NOMBRE.
      VRFC = MOVTO.RFC.
      FIND HORARIO WHERE HORARIO.COD-PROG = VCOD-PROG AND
        HORARIO.CVE-FUN = MOVTO.CVE-FUN NO-ERROR.

```

```

IF AVAILABLE HORARIO THEN DO:
  IF MOVTO.TIPO-MOVTO = "ALTA" THEN DO:
    VALTA = "_X".
    IF MOVTO.CVE-CAUSA = "11" THEN VNVOINGR = "_X".
    IF MOVTO.CVE-CAUSA = "12" THEN VREINGR = "_X".
    IF MOVTO.CVE-CAUSA = "13" THEN VAUMHOR = "_X".
    IF MOVTO.CVE-CAUSA = "14" THEN VREANUD = "_X".
    IF MOVTO.CVE-CAUSA = "15" THEN VINTER = "_X".
  END.
  ELSE DO:
    IF MOVTO.TIPO-MOVTO = "BAJA" THEN DO:
      VBAJA = "_X".
      IF MOVTO.CVE-CAUSA = "21" THEN VOTRON = "_X".
      IF MOVTO.CVE-CAUSA = "22" THEN VRECLAS = "_X".
      IF MOVTO.CVE-CAUSA = "23" THEN VTRANSF = "_X".
      IF MOVTO.CVE-CAUSA = "24" THEN VDISHOR = "_X".
      IF MOVTO.CVE-CAUSA = "25" THEN VENFERM = "_X".
      IF MOVTO.CVE-CAUSA = "26" THEN VGRAVI = "_X".
      IF MOVTO.CVE-CAUSA = "27" THEN VSUSPEN = "_X".
      IF MOVTO.CVE-CAUSA = "28" THEN VRENUN = "_X".
      IF MOVTO.CVE-CAUSA = "29" THEN VRESIC = "_X".
      IF MOVTO.CVE-CAUSA = "30" THEN VDEFUN = "_X".
    END.
    ELSE DO:
      IF MOVTO.TIPO-MOVTO = "LICENCIA" THEN DO:
        VLIC = "_X".
        IF MOVTO.CVE-CAUSA = "31" THEN VANIOS = "_X".
        IF MOVTO.CVE-CAUSA = "32" THEN VCOMOFIC = "_X".
        IF MOVTO.CVE-CAUSA = "33" THEN VESTPAIS = "_X".
        IF MOVTO.CVE-CAUSA = "34" THEN VESTEXT = "_X".
        IF MOVTO.CVE-CAUSA = "35" THEN VPARTIC = "_X".
      END.
    END.
  END.
  IF HORARIO.DIA-1 = "LU" THEN VLU1=HORARIO.HOR-INI1.
  IF HORARIO.DIA-1 = "LU" THEN VLU2=HORARIO.HOR-FIN1.
  IF HORARIO.DIA-1 = "MA" THEN VMA1=HORARIO.HOR-INI1.
  IF HORARIO.DIA-1 = "MA" THEN VMA2=HORARIO.HOR-FIN1.
  IF HORARIO.DIA-1 = "MI" THEN VMI1=HORARIO.HOR-INI1.
  IF HORARIO.DIA-1 = "MI" THEN VMI2=HORARIO.HOR-FIN1.
  IF HORARIO.DIA-1 = "JU" THEN VJU1=HORARIO.HOR-INI1.
  IF HORARIO.DIA-1 = "JU" THEN VJU2=HORARIO.HOR-FIN1.
  IF HORARIO.DIA-1 = "VI" THEN VVI1=HORARIO.HOR-INI1.
  IF HORARIO.DIA-1 = "VI" THEN VVI2=HORARIO.HOR-FIN1.

```

```

IF HORARIO.DIA-2 = "LU" THEN VLU1=HORARIO.HOR-INI2.
IF HORARIO.DIA-2 = "LU" THEN VLU2=HORARIO.HOR-FIN2.
IF HORARIO.DIA-2 = "MA" THEN VMA1=HORARIO.HOR-INI2.
IF HORARIO.DIA-2 = "MA" THEN VMA2=HORARIO.HOR-FIN2.
IF HORARIO.DIA-2 = "MI" THEN VMI1=HORARIO.HOR-INI2.
IF HORARIO.DIA-2 = "MI" THEN VMI2=HORARIO.HOR-FIN2.
IF HORARIO.DIA-2 = "JU" THEN VJU1=HORARIO.HOR-INI2.
IF HORARIO.DIA-2 = "JU" THEN VJU2=HORARIO.HOR-FIN2.
IF HORARIO.DIA-2 = "VI" THEN VVI1=HORARIO.HOR-INI2.
IF HORARIO.DIA-2 = "VI" THEN VVI2=HORARIO.HOR-FIN2.
IF HORARIO.CVE-CATEG = "1A" THEN VCATEGORIA="AYUDANTE A".
IF HORARIO.CVE-CATEG = "1B" THEN VCATEGORIA="AYUDANTE B".
IF HORARIO.CVE-CATEG = "2A" THEN VCATEGORIA="ASIGNATURA A".
IF HORARIO.CVE-CATEG = "2B" THEN VCATEGORIA="ASIGNATURA B".
IF HORARIO.CVE-CATEG = "3A" THEN VCATEGORIA="ASIGNATURA A
DEFIN".
IF HORARIO.CVE-CATEG = "3B" THEN VCATEGORIA="ASIGNATURA B
DEFIN".
SUMTEO = HORARIO.HOR-FIN1 - HORARIO.HOR-INI1 +
HORARIO.HOR-FIN2 - HORARIO.HOR-INI2.
SUMPRA = HORARIO.HOR-FIN1 - HORARIO.HOR-INI1 +
HORARIO.HOR-FIN2 - HORARIO.HOR-INI2.
IF HORARIO.ACTIVI = "CLINICA" THEN VPRACTICA = SUMPRA.
IF HORARIO.ACTIVI = "TEORIA" THEN VTEORIA = SUMTEO.
IF HORARIO.ACTIVI = "LABORATORIO" THEN VPRACTICA = SUMPRA.
IF HORARIO.ACTIVI = "BIBLIOGRAF" THEN VTEORIA = SUMTEO.
IF HORARIO.ACTIVI = "SEMINARIO" THEN VTEORIA = SUMTEO.
IF HORARIO.ACTIVI = "COMUNITARIA" THEN VPRACTICA = SUMPRA.
IF HORARIO.ACTIVI = "PRACTICA IMSO" THEN VPRACTICA = SUMPRA.
END.
OUTPUT TO PRINTER PAGE-SIZE 51.
VIEW FRAME ENCAB.
DISPLAY VCOD-PROG HORARIO.GRUPO VLU1 VLU2 VMA1 VMA2 VMI1
VMI2 VJU1 VJU2 VVI1 VVI2 HORARIO.CVE-FUN HORARIO.CVE-CAT
VCATEGORIA VTEORIA VPRACTICA MOVTO.FECH-MOVTO
MOVTO.VIGENCIA
WITH FRAME DETALLE.
END.
END.
END.
OUTPUT CLOSE.
VNO-MOVTO = " ".
VCOD-PROG = " ".
END.

```

/\* <HORARIO.P> menu para elaborar horarios de alumnos y profesores \*/

```
DEF VAR J AS INTEGER.
DEF VAR FES AS CHAR FORMAT "X(13)" INITIAL "FES ZARAGOZA".
DEF VAR MENU1 AS CHAR EXTENT 2 FORMAT "X(22)"
  INITIAL ["1. HORARIO DE GRUPO",
    "2. HORARIO DE PROFESOR"].
DEF VAR PROGRAMA AS CHAR EXTENT 8 FORMAT "X(10)"
  INITIAL ["HORAGRUP.P","HORAPROF.P"].
```

FORM HEADER

```
FES AT 2 "MENU DE ELABORACION DE HORARIOS" AT 27 TODAY AT 69
FORMAT "99-99-9999" WITH PAGE-TOP WIDTH 80 TITLE "S C O H P"
CENTER FRAME TITULO ROW 2.
```

REPEAT:

```
HIDE ALL.
VIEW FRAME TITULO.
PUT SCREEN.ROW 7 COLUMN 37 COLOR MESSAGE " MENU ".
DISPLAY MENU1 WITH NO-LABEL FRAME MEN1 1 COLUMN CENTER ROW 10.
MESSAGE
" *" + CHR(17) + CHR(196) + CHR(196) + CHR(217) + "SELECCIONAR OPCION
*" + CHR(24) + " ANTERIOR  *" + CHR(25) + " SIGUIENTE *F4* SALIR ".
CHOOSE FIELD MENU1 AUTO-RETURN WITH FRAME MEN1.
IF CAN-DO("RETURN,PICK,GO",KEYFUNCTION(LASTKEY)) THEN
DO J= 1 TO 3:
  IF FRAME-VALUE = MENU1[J] THEN LEAVE.
END.
ELSE DO:
  BELL.
  NEXT.
END.
HIDE MESSAGE.
PAUSE 0.
HIDE ALL.
IF SEARCH (PROGRAMA[J]) = ? THEN
  MESSAGE "EL PROGRAMA: " PROGRAMA[J] "NO EXISTE".
ELSE RUN VALUE (PROGRAMA[J]).
END.
/*QUIT*/
```