

EGE00000345

LB

1028.73

G3

1007

SISTEMA DE TUTOREO INTELIGENTE PARA  
EL APRENDIZAJE DE LAS LEYES DE LOS  
EXONENTES



TESIS

MAESTRIA EN CIENCIAS  
ESPECIALIDAD EN TECNOLOGIA INFORMATICA

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS  
SUPERIORES DE MONTERREY  
CAMPUS MONTERREY

POR

BENIAMIN EDUARDO GARCIA VILLEGAS

DICIEMBRE DE 1997

**SISTEMA DE TUTOREO INTELIGENTE PARA EL APRENDIZAJE  
DE LAS LEYES DE LOS EXPONENTES**



**TESIS**

**MAESTRÍA EN CIENCIAS  
ESPECIALIDAD EN TECNOLOGÍA INFORMÁTICA**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS  
SUPERIORES DE MONTERREY  
CAMPUS MONTERREY**

**POR**

**BENJAMÍN EDUARDO GARCÍA VILLEGAS**

**DICIEMBRE DE 1997**

**SISTEMA DE TUTOREO INTELIGENTE PARA EL APRENDIZAJE  
DE LAS LEYES DE LOS EXPONENTES**

**POR**

**BENJAMÍN EDUARDO GARCÍA VILLEGAS**

**TESIS**

**Presentada a la División de Graduados en  
Computación, Información y Comunicaciones  
Este trabajo es requisito parcial para obtener el título de Maestro  
en Ciencias con especialidad en Tecnología Informática.**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS  
SUPERIORES DE MONTERREY  
CAMPUS MONTERREY**

**DICIEMBRE DE 1997**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS  
SUPERIORES DE MONTERREY  
CAMPUS MONTERREY**

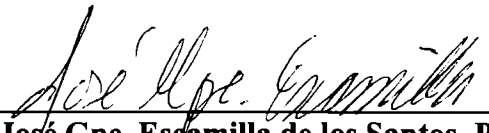
**DIVISIÓN DE GRADUADOS EN COMPUTACIÓN, INFORMACIÓN Y  
COMUNICACIONES**

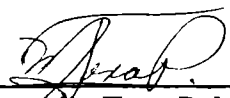
**PROGRAMAS DE POSGRADO EN COMPUTACIÓN, INFORMACIÓN Y  
COMUNICACIONES**


**Los miembros del comité de tesis recomendamos que la presente tesis del Ing.  
Benjamín Eduardo García Villegas sea aceptada como requisito parcial para obtener  
el grado académico de Maestro en Ciencias, especialidad en:**

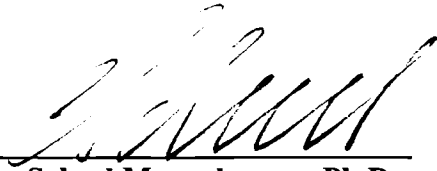
**TECNOLOGÍA INFORMÁTICA**

**Comité de tesis:**

  
\_\_\_\_\_  
**José Gpe. Escamilla de los Santos, Ph.D.  
Asesor Principal**

  
\_\_\_\_\_  
**Margarita Toro Palacios, M.C.  
Sinodal**

  
\_\_\_\_\_  
**Moraima Campbell Dávila, M.C.A.  
Sinodal**

  
\_\_\_\_\_  
**Carlos Scheel Mayenberger, Ph.D.  
Director de los Programas de Posgrado en  
Computación, Información y Comunicaciones.**

**DICIEMBRE DE 1997**

# Dedicatoria.

A *Dios*, por estar siempre a mi favor, por mantenerme saludable durante esta lucha y por escucharme y acompañarme en los momentos difíciles. Nunca me abandones Padre.

A mis padres *Ramón* y *Norma*, a quienes debo lo que soy, por haberme hecho un hombre con principios, por enseñarme la honradez, la humildad y el amor a Dios, y por apoyarme y amarme siempre.

A toda *mi familia*, por su apoyo desinteresado, sus palabras de ánimo y por la fe que siempre han depositado en mi persona.

A mi novia *Nelia*, por su alegre compañía, su apoyo en todo momento y su amor.

A todos aquellos que reconocen la presencia divina en el amor humano. Nunca desfallezcan en su lucha constante pues nada de lo que han logrado ha sido en vano.

A todos ellos mi cariño y respeto.

Benjamín Eduardo García Villegas

# Reconocimientos.

Quiero expresar mi más sincero agradecimiento a todas aquellas personas que de alguna manera contribuyeron al logro de este trabajo de tesis y que sin su apoyo no hubiera sido posible su realización:

A mi comité de tesis: *Dr. José Gpe. Escamilla de los Santos, Margarita Toro Palacios y Moraima Campbell Dávila*, por sus comentarios, observaciones y sugerencias que permitieron enriquecer esta investigación.

A mis amigos: *Alonso Pérez, Claudia Morales, Juan García, Javier Carlos, Rogelio Robledo, Miguel Wister, Laura Salinas, Pablo Pancardo, Angeles Constantino, Jorge Navarro, Myriam Rochin, Ana María Sánchez*, por su sincera amistad, sus palabras de ánimo y su apoyo incondicional.

A mis *maestros* por sus enseñanzas y por guiarme con su sabiduría durante la realización de esta etapa en mi desarrollo personal.

Asi también, deseo agradecer a las instituciones que me brindaron su apoyo para la conclusión de mis estudios de maestría.

Al *ITESM* por abrirme las puertas de sus aulas y darme la oportunidad de subir un escalón más en mi vida académica.

Al *Centro de Inteligencia Artificial* del ITESM por las facilidades prestadas para el desarrollo de esta investigación.

Al *CONACYT* por su apoyo económico durante el periodo de mis estudios en Monterrey.

Mil gracias.

Benjamín Eduardo García Villegas

# Resumen.

Los errores que comete el estudiante en el estudio de las matemáticas son parte del proceso de conocimiento; la ciencia, el conocimiento de lo verdadero, es un proceso continuo de corrección de errores. Los estudios en esta área aunado a los avances tecnológicos permiten la detección de las causas de los errores de manera automática a través del uso de sistemas de tutorio inteligentes. Los tutores inteligentes se caracterizan por la aplicación de técnicas de Inteligencia Artificial, contienen un cúmulo de conocimiento sobre un área específica, que los hace aptos para resolver problemas y hacer inferencias, llegan a deducir aproximadamente las habilidades, conocimientos y deficiencias del estudiante que con él interactúa y ponen en práctica una estrategia instruccional

El presente trabajo describe el diseño de un sistema de tutorio inteligente para la simplificación de expresiones algebraicas mediante las cuatro Leyes de los Exponentes que se enseñan en el nivel preparatoria en el sistema ITESM. Presenta los conceptos y técnicas relativas a los principales módulos que conforman un sistema de tutorio inteligente: el *módulo experto*, el *módulo del estudiante*, *módulo instruccional*, e *interfaz de usuario*. Entre las ventajas obtenidas con el desarrollo de este sistema se encuentran las siguientes: la sencillez de implementación, la capacidad de diagnosticar el desempeño del estudiante y, la capacidad de fungir como un medio para la práctica de ejercicios de simplificación de expresiones, en donde el usuario recibe retroalimentación del sistema.

Entre los logros alcanzados durante la realización de este trabajo sobresalen: (1) el sistema está apoyado en una técnica de modelación del estudiante que es una variante de la técnica del *catálogo de errores* con asignación de pesos difusos a cada elemento de la base de conocimientos, (2) la integración de diversos estudios teóricos en el área de diseño de sistemas de tutorio inteligente, (3) el diseño de un conjunto de reglas de simplificación de expresiones para su uso en sistemas expertos, (4) el diseño de reglas de simplificación que simulan errores que más comúnmente cometen los estudiantes durante su aprendizaje y, (5) la obtención de un primer bosquejo para la creación de una interfaz de usuario apropiada a los propósitos del sistema y las características de los usuarios potenciales del mismo. Este sistema ha sido desarrollado en Clips, un Shell para la construcción de sistema expertos con razonamiento hacia adelante producido por la NASA.

Es preciso la creación de sistemas computacionales que faciliten, apoyen y mejoren el proceso de enseñanza aprendizaje, es por eso que la creación de un sistema de Tutorio Inteligente sobre las Leyes de los Exponentes constituye un paso muy importante hacia esta meta dentro de las preparatorias en el sistema ITESM, como soporte para el desarrollo de las habilidades de resolución de problemas, y el aprovechamiento de las ventajas del uso de los errores del estudiante como base para construir en él el conocimiento correcto.



# Índice.

Dedicatoria .....	iv
Reconocimientos.....	v
Resumen .....	vi
Índice .....	vii
Lista de figuras .....	ix
Capítulo 1. Introducción .....	1
1.1. Las dificultades en el aprendizaje de las matemáticas .....	1
1.2. Los Sistemas de Tutorío Inteligentes .....	2
1.3. Resumen del contenido del escrito .....	4
Capítulo 2. El Problema .....	5
2.1. Propósito del proyecto .....	5
2.2. Contribución .....	5
2.3. Definición del problema .....	6
2.3.1. Metas .....	6
2.3.2. Delimitaciones .....	7
2.4. Producto final .....	7
2.5. Productos secundarios .....	7
2.6. Hipótesis .....	8
Capítulo 3. Estado del arte .....	9
3.1. Módulos de los sistemas de tutorío inteligente .....	9
3.1.1. Módulo experto .....	9
3.1.2. Módulo del estudiante .....	12
3.1.3. Módulo instruccional .....	14
3.1.4. Ambiente instruccional .....	16
3.1.5. Interfaz .....	17
3.2. Aplicaciones similares .....	19
3.2.1. El Practical Algebra Tutor (PAT) .....	19
Capítulo 4. Diseño General del Sistema .....	23
4.1. Diseño Funcional .....	23

4.1.1. Prerrequisitos y necesidades .....	23
4.1.2. Especificación funcional .....	24
4.1.3. Actividades de aprendizaje .....	26
4.2. Diseño de la interfaz .....	28
4.2.1. Metodología de diseño .....	28
4.2.2. Modelo del diseñador acerca de la actividad .....	29
4.2.3. Modelo del diseñador acerca del usuario potencial .....	30
4.2.4. Diseño de Interacción .....	32
4.2.5. Metáforas que propone el diseño .....	33
4.2.6. Diagrama de Navegación del Diseño propuesto .....	34
4.2.7. Diseño de las pantallas del sistema .....	34
<b>Capítulo 5. Diseño de los módulos inteligentes .....</b>	<b>35</b>
5.1. Módulo Experto .....	36
5.1.1. Simplificación de expresiones .....	36
5.1.2. Especificación de reglas correctas de simplificación .....	37
5.2. Módulo del estudiante .....	43
5.2.1. Modelo del estudiante .....	43
5.2.2. Especificación de reglas incorrectas .....	47
5.3. Módulo instruccional .....	50
5.3.1. Selección de ejercicios y niveles de dificultad .....	50
5.3.2. Consejos para el estudiante .....	51
<b>Capítulo 6. Conclusiones y trabajos futuros .....</b>	<b>53</b>
6.1. Conclusiones .....	53
6.2. Trabajos futuros .....	55
<b>Anexos .....</b>	<b>58</b>
<b>Bibliografía .....</b>	<b>81</b>
<b>Vita .....</b>	<b>84</b>

# Lista de figuras.

<b>Figura 1-1.</b> Interacción entre los módulos de un Sistema de Tutorio Inteligente.....	3
<b>Figura 3-1.</b> Relación entre <i>efectividad pedagógica</i> y <i>esfuerzo de implementación</i> .....	11
<b>Figura 3-2.</b> Panorama del módulo experto.....	12
<b>Figura 3-3.</b> Las tres dimensiones del modelo del estudiante .....	14
<b>Figura 3-4.</b> Panorama para el diseño del módulo instruccional .....	16
<b>Figura 3-5.</b> Aspectos para especificar el diseño de la interfaz .....	19
<b>Figura 3-6.</b> Pantalla del sistema PAT que presenta una descripción escrita de una situación.....	20
<b>Figura 3-7.</b> Representación gráfica del problema usando el sistema PAT .....	21
<b>Figura 3-8.</b> Representación tabular del problema usando el sistema PAT .....	21
<b>Figura 4-1.</b> Diagrama de bloques que describe la funcionalidad global del sistema .....	24
<b>Figura 4-2.</b> Diagrama de flujo de la <i>sesión de ejercicios</i> .....	27
<b>Figura 4-3.</b> Diagrama de navegación.....	34
<b>Figura 5-1.</b> Estructura principal del sistema .....	35
<b>Figura 5-2.</b> Reglas de simplificación de expresiones .....	42
<b>Figura 5-3.</b> Relaciones entre reglas correctas e incorrectas .....	45
<b>Figura 5-4.</b> Reglas simplificación incorrectas.....	49
<b>Figura 5-5.</b> Procedimiento para la selección de ejercicios .....	51
<b>Figura 5-6.</b> Tabla que muestra algunos de los consejos presentados por el sistema.....	52

# Capítulo 1. Introducción.

## 1.1. Las dificultades en el aprendizaje de las matemáticas.

Los errores que comete el estudiante en el estudio de las matemáticas son parte del proceso de conocimiento; la ciencia, el conocimiento de lo verdadero, es un proceso continuo de corrección de errores. Los errores son, de hecho, un objeto de estudio para la educación matemática, es necesaria su detección, valoración y corrección en el estudiante.

Cuando hablamos de los errores en el aprendizaje de las matemáticas, el aspecto central a considerar es cuántos y qué tipo de conceptos están involucrados en el error, cuántas piezas de conocimiento contiene, lo cual es la tarea fundamental de un *análisis conceptual*. El paso más conveniente para estudiar los errores y las dificultades de las matemáticas es realizar un análisis conceptual de todos los elementos de cada una de las nociones fundamentales del currículo de las matemáticas [Rico, 1993].

En Francia se trabaja con la noción del *obstáculo epistemológico*, tomado de los estudios y reflexiones de Gaston Bachelard [Bachelard, 1988], se busca el análisis de la evolución histórica de los conceptos matemáticos, es decir, la detección de los errores y las concepciones incorrectas que han surgido durante la evolución histórica de esos conceptos.

Bachelard expone que *hay que plantear el problema del conocimiento científico en términos de obstáculos*, pero no de obstáculos externos, sino de obstáculos que aparecen en el momento mismo de conocer. El estudio del obstáculo epistemológico es útil para el desarrollo histórico del pensamiento científico y para la práctica educativa.

“El historiador de la ciencia debe tomar las ideas como hechos. El epistemólogo debe tomar los hechos como ideas, insertándolas en un sistema de pensamientos. Un hecho mal

interpretado por una época, sigue siendo un *hecho* para el historiador. Según el epistemólogo es un *obstáculo*, un contratiempo.” [Bachelard,1988].

El estudiante no es como una tabla rasa sobre la cual se empieza a escribir; posee conocimientos de experiencias pasadas que condicionan la adquisición del nuevo conocimiento.

Los errores pues, no surgen por azar, se basan en conocimientos previos, los cuales es necesario considerar durante el proceso de aprendizaje ya que pueden contribuir positivamente en el proceso de aprendizaje [Rico, 1993].

Durante el proceso de enseñanza el maestro debe saber reconocer los errores en el conocimiento de los estudiantes. Para ésto, el maestro usa su experiencia en el área que él enseña y su experiencia en reconocer los errores más comunes de los estudiantes. La experiencia del maestro en estos dos tópicos puede llevarse a un sistema computacional inteligente, el cual queda facultado para realizar estas dos operaciones en forma automática cuando interactúa con un usuario. De esta forma podemos lograr la creación de un sistema capaz de enseñar y reconocer errores en los estudiantes, o mejor dicho, un *Sistema de Tutorio Inteligente*.

## **1.2. Los Sistemas de Tutorio Inteligentes.**

Los Sistemas de Tutorio Inteligente (STI) representan una nueva etapa en la tradicional Instrucción Asistida por Computadora (CAI), es decir, los sistemas de tutorio que no hacen uso de la Inteligencia Artificial. La creación de los CAI inició en 1960, se caracterizan por la presentación de material instruccional con animaciones, gráficas e información en general al estudiante y la realización evaluaciones para que el estudiante avance y le sea presentado material correspondiente a un conocimiento más complejo o profundo sobre el tema o área del sistema [1983, Chambers].

Los STI toman todas las características de los CAI, sólo que ahora presentan material, ofrecen ayuda, sugieren ejercicios y evalúan al estudiante, aplicando técnicas de Inteligencia Artificial, lo que produce una instrucción más individualizada. Los STI se caracterizan por la aplicación de técnicas de Inteligencia Artificial en tres aspectos. Primero, contienen un cúmulo de conocimiento sobre un área específica, que los hace aptos para resolver problemas e inferir nuevo conocimiento. Segundo, llegan a deducir aproximadamente las habilidades, conocimientos y deficiencias del estudiante que con él interactúa. Tercero, ponen en práctica una estrategia instruccional para reducir la diferencia de conocimientos y habilidades entre el estudiante y su conocimiento almacenado [Burns y Capps, 1988].

Por su funcionalidad, es común estudiar y diseñar los sistemas de tutorio inteligente basándose en los tres aspectos arriba mencionados. Los investigadores en el área hablan de

cinco módulos: experto, del estudiante, pedagógico, ambiente instruccional e interfaz, los cuales interactúan de la manera descrita en el diagrama de la figura 1-1.

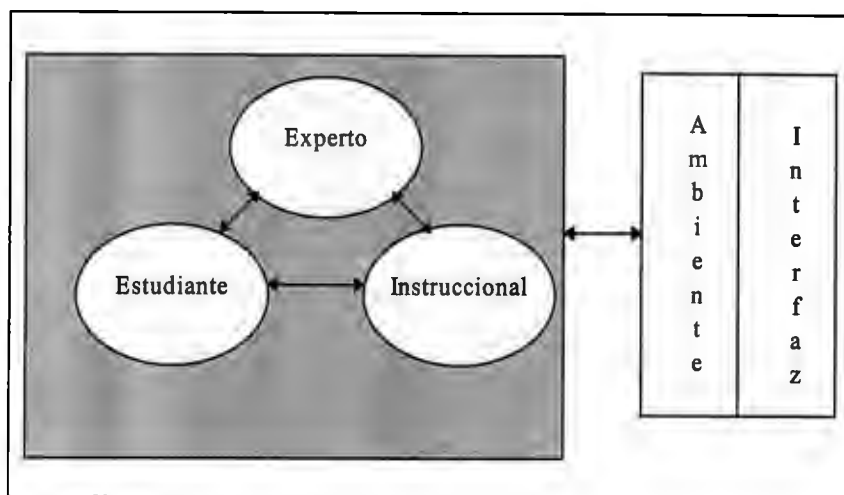


Figura 1-1. Interacción entre los módulos de un Sistema de Tutorio Inteligente.

El *módulo experto* cumple una doble función dentro de los tutores inteligentes; por un lado, proporciona el conocimiento en el dominio particular, y por el otro, representa el estándar para la evaluación del estudiante. La elección del modelo para la representación del conocimiento es el aspecto más importante en el diseño del módulo experto. Debe ser el que mejor permita la solución de problemas e inferencias en el dominio específico del conocimiento.

El *módulo del estudiante* determina qué sabe y qué no sabe el estudiante. Para llegar a conocer esto, el módulo puede apoyarse en: (1) el desempeño del estudiante al intentar resolver problemas, (2) las preguntas directas hechas al estudiante, (3) los datos históricos que permitan medir su nivel de habilidad, y (4) el nivel de dificultad.

El *módulo instruccional* es el encargado de diseñar la estrategia instruccional de acuerdo a las valoraciones sobre el estudiante. Este módulo identifica las diferencias de conocimiento entre el estudiante y el experto para luego seleccionar una estrategia instruccional adecuada. Además, puede estar provisto con capacidad para responder a las preguntas del estudiante y ayudarlo cuando enfrente dificultades [Burns y Capps, 1988].

El *ambiente instruccional* está conformado por todas aquellas herramientas, situaciones y actividades que apoyan y facilitan el aprendizaje y todo lo que hace estudiante al interactuar con el sistema.

La *interfaz* es el módulo de comunicación entre el sistema y el usuario. Su manejo debe ser sencillo al usuario, ya que de no ser así, éste tendrá dos problemas, mejorar sus conocimientos en el área de interés, y además, aprender el manejo del sistema. Para lograr

lo anterior puede hacerse uso de medios tales como presentación gráfica, hipertexto, o bien técnicas de lenguaje natural, o alguna combinación de éstos, dependiendo de las características del usuario.

### **1.3. Resumen del contenido del escrito.**

El capítulo dos de este escrito describe el problema planteado para este trabajo de tesis, los estudios que motivaron a su planteamiento, las aportaciones más importantes, las metas, las limitaciones y el producto final resultado de este trabajo.

En el capítulo tres se presenta el estado del arte en el área de los sistemas de tutorio inteligente. Se describe cada uno de los módulos que lo componen, las técnicas más importantes empleadas en su implementación y algunos sistemas que se han desarrollado probando la eficacia de estas técnicas.

El capítulo cuatro muestra el diseño funcional propuesto para el sistema, el estudio seguido para el diseño de la interfaz de usuario. Se presentan diagramas y descripciones sobre la manera en que interactúan los diferentes componentes que lo constituyen, el diagrama de navegación, y un primer estudio sobre los usuarios potenciales del sistema.

En el capítulo cinco se describen en detalle los módulos inteligentes del sistema, las técnicas que se emplearon en su diseño y se muestra la relación que guardan con algunas de las técnicas presentadas en el estado del arte en el capítulo 3. Este capítulo trata entre otras cosas sobre el módulo experto para la simplificación de expresiones, la técnica empleada para la modelación del estudiante y, la selección de ejercicios.

El capítulo seis contiene las conclusiones derivadas del desarrollo del presente trabajo, así como una serie de sugerencias que podrían plantearse como trabajos futuros para mejorar lo hasta ahora alcanzado.

## **Capítulo 2. El Problema.**

### **2.1. Propósito del proyecto.**

La tesis planteada en el presente trabajo tiene como propósito diseñar un Sistema de Tutorio Inteligente que apoye el desarrollo de habilidades en la manipulación de las Leyes de los Exponentes como herramienta para la práctica de ejercicios de simplificación de expresiones algebraicas y como herramienta de diagnóstico de los conocimientos del alumno en la solución de ejercicios en este tema particular de las matemáticas.

### **2.2. Contribución.**

1. Integrar una serie de estudios sobre el diseño de los diferentes módulos de un sistema de tutorio inteligente, como:
  - detección de las causas de errores en el álgebra,
  - modelación del conocimiento del estudiante,
  - simplificación de expresiones algebraicas,
  - diseño de la interfaz de usuario, etc.
2. Construcción de un primer prototipo del sistema propuesto.

Además, el uso de un Sistema de Tutorio Inteligente permitirá una instrucción más individualizada y un mejor diagnóstico del desempeño del estudiante. Lo anterior significa una detección más oportuna y precisa de las dificultades que afronta el alumno al estudiar el tema, información que servirá al maestro y al alumno para que centren su atención a aminorar tales dificultades.



## **2.3. Definición del problema.**

De acuerdo con Kilpatrick [Kilpatrick, 1993], las matemáticas dentro del currículo escolar se distinguen por ser el tema de mayor dificultad, fracaso y rechazo por el estudiante durante el proceso de enseñanza aprendizaje, pero poseen una innegable importancia dentro del currículo escolar. Hoy en día, en la mayoría de los países se enfatiza el desarrollo del razonamiento de las habilidades de resolución de problemas sobre la memorización de hechos y procedimientos. Además, los errores tienen una enorme importancia para la construcción y consolidación del conocimiento del estudiante; Popper dice que el conocimiento no puede partir de la nada. El avance en el conocimiento es en gran medida la modificación de conocimiento anterior, [Rico, 1993]. Por otro lado, apoyar el estudio y la enseñanza de las matemáticas en la tecnología, no significa el reemplazo de los profesores. Como diría Kilpatrick textualmente:

“Los diseñadores de la política educativa continúan presentándonos la imagen futurista de un salón de clases lleno de tecnología en el que profesor ha sido reemplazado por un programa de enseñanza por computador. No obstante, la investigación ha mostrado repetidamente que ésta es una imagen errada. La atención de los investigadores y de los desarrolladores en tecnología computacional se ha centrado ahora en el desarrollo de programas de computador para la enseñanza, algunos para profesores y otros para los alumnos, pero todos manejados por un profesor y no por un técnico.” [Kilpatrick, 1993]

Los maestros y alumnos al igual que el resto de los sectores de la sociedad no pueden quedar al margen de los beneficios del uso de los sistemas computacionales, ya no sólo como un apoyo a sus actividades, sino dentro del mismo proceso de enseñanza aprendizaje. Es preciso la creación de sistemas computacionales que faciliten, apoyen y mejoren el proceso de enseñanza aprendizaje, es por eso que la creación de un sistema de Tutorío Inteligente sobre las Leyes de los Exponentes constituye un paso muy importante hacia esta meta dentro del sistema de preparatorias en el ITESM, como soporte para el desarrollo del razonamiento de las habilidades de resolución de problemas, y el aprovechamiento de las ventajas del uso de los errores del estudiante como base para construir en él el conocimiento correcto.

### **2.3.1. Metas.**

- Ofrecer al maestro la posibilidad de una herramienta de diagnóstico del desempeño del estudiante.

- Ofrecer a los estudiantes la posibilidad de un medio novedoso para practicar ejercicios de simplificación de expresiones, recibir retroalimentación y, un diagnóstico de sus dificultades.

### **2.3.2. Delimitaciones.**

Se pretende que en el diseño resultante:

- El conocimiento contenido en el *módulo experto* resuelva problemas de la complejidad necesaria para la enseñanza del tema de las Leyes de los Exponentes.
- El *módulo instruccional* cumplirá la función de aconsejar al estudiante y al maestro de acuerdo a los problemas detectados en el estudiante, y no diseñará estrategia instruccional alguna más compleja.
- La interfaz diseñada, será una propuesta surgida de un primer estudio en el conocimiento del usuario y su actividad, y que podrá mejorarse en una continuación de este trabajo, ya que el diseño de interfaz es proceso iterativo de prueba y mejora.

### **2.4. Producto final.**

El producto final resultado de este trabajo de tesis es el diseño de un Sistema de Tutorio Inteligente que posibilite la práctica de ejercicios de simplificación de expresiones algebraicas y permita diagnosticar el desempeño de los estudiantes que interactúen con él.

### **2.5. Productos secundarios.**

- Creación de un primer prototipo de acuerdo al diseño propuesto,
- Un estudio inicial del usuario para la creación de la interfaz adecuada,
- Creación de un simplificador de expresiones algebraicas con las limitaciones de conocimiento descritas en la sección de *delimitaciones* de éste capítulo.

## **2.6. Hipótesis.**

Es posible diseñar un Sistema de Tutorio Inteligente en el área de las Leyes de los Exponentes que integre diferentes estudios sobre Tutores Inteligentes y que sea capaz de diagnosticar los conocimientos del estudiante, funcione como una herramineta de práctica de ejercicios y ofrezca a los maestros una fuente de información acerca del estado del conocimiento del estudiante.

## Capítulo 3. Estado del arte.

En el capítulo uno se dio una descripción general de los sistemas de tutorio inteligente, de sus componentes y de la manera en que interactúan entre ellos. El presente capítulo presenta en una forma más detallada cada uno de los módulos que constituyen un tutor inteligente. El capítulo está organizado de acuerdo a los módulos más importantes del tutor inteligente, y describe para cada uno de ellos las técnicas más sobresalientes y ejemplos de aplicaciones. Hacia el final del capítulo se detallan las características del Practical Algebra Tutor (PAT), una de las aplicaciones más recientes de los tutores inteligentes en el área de las matemáticas.

### 3.1. Módulos de los sistemas de tutorio inteligente.

Tradicionalmente los investigadores han dividido a los Sistemas de Tutores Inteligentes (STI's) en módulos que interactúan entre sí para cumplir su función. La mayoría de los investigadores coincide en dividir el STI en *módulo experto*, *módulo del estudiante*, *módulo instruccional*, *ambiente instruccional* e *interfaz de usuario*.

#### 3.1.1. Módulo experto.

El módulo experto es el que almacena el conocimiento sobre algún área específica. La experiencia en la construcción de STI's dicta que el módulo experto debe contener una gran cantidad de conocimiento y que éste debe estar lo suficientemente detallado para que el sistema sea capaz de realizar la función de instruir. Una fase importante en el diseño y desarrollo del módulo experto lo constituye la adquisición del conocimiento, esto es, la extracción del conocimiento de algún experto en el área de aplicación del STI. Esta tarea involucra entrevistas con el experto para obtener su conocimiento y la manera en que lo aplica para luego transformarlo a una forma útil para el sistema STI. Es recomendable que

la persona o grupo de personas seleccionadas como expertos deben tener al menos 10 años de experiencia en el dominio de interés. Además, la elección del modelo para codificar el conocimiento se hace en base a los propósitos del STI. Se han desarrollado tres clases de modelos para la representación del conocimiento: de *caja negra*, de *caja de cristal* y *modelos cognitivos* [Anderson, 1988].

El modelo de representación de *caja negra* es muy utilizado para el diseño de los sistemas expertos y se distingue por ser capaz de ofrecer resultados correctos a los problemas o preguntas del dominio del sistema, pero es incapaz de dar una explicación adecuada de sus conclusiones; esto trae como consecuencia que sea poco apropiado para la instrucción. Sin embargo, puede emplearse en STI's que den consejos apropiados al estudiante sobre el comportamiento correcto en una tarea o habilidad. Una alternativa a esta limitación es la aplicación de una metodología conocida como tutorio basado en temas [Burton y Brown, 1982]. La idea consiste en la identificación de patrones sobre el comportamiento del estudiante y del experto y basar la instrucción en estos patrones. Un ejemplo del uso del modelo de caja negra es el sistema WEST, un juego para mejorar las habilidades aritméticas [Burton y Brown, 1982]. Cuando el estudiante no realiza la misma acción que el experto, se le ofrece un consejo.

El segundo tipo de representación es el modelo de *caja de cristal*. Esta clase de representación es similar a la manera en que los seres humanos almacenamos el conocimiento, aunque difiere en la forma como lo usamos. Para su construcción, se requiere la intervención de ingenieros del conocimiento y expertos en el área. La codificación comúnmente se hace usando reglas de producción y su estructura los hace prácticos en el diseño de STI's. El ejemplo clásico de este tipo de representación es el sistema GUIDON [Clancey, 1982], que es un STI basado en el sistema experto MYCIN [Shortliffe, 1976], orientado a mejorar las habilidades de diagnóstico de infecciones bacteriales.

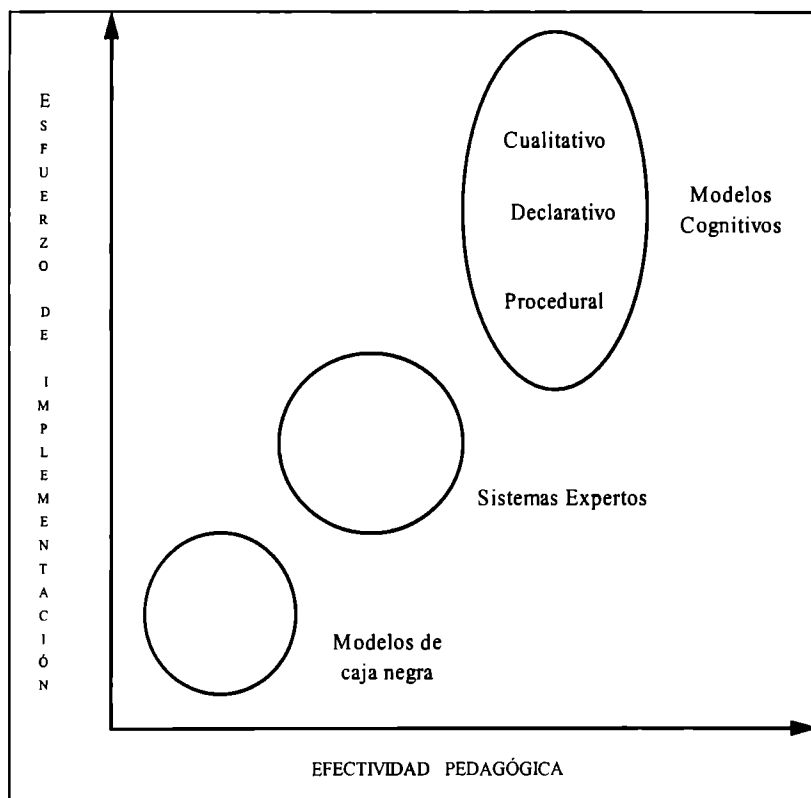
Como mencionamos en el párrafo anterior, un módulo experto diseñado con un modelo de caja de cristal almacena el conocimiento como los humanos, pero razona de una forma muy diferente. Una tercera alternativa que rompe con esta restricción es el *modelo cognitivo*, el cual intenta usar el conocimiento con las limitaciones de un humano. Para lograr esto, el modelo de representación debe contener una gran cantidad de conocimiento muy bien detallado, lo que significa una ardua labor en la adquisición y representación del conocimiento. Este tipo de modelo es el más apropiado para el tutorio, pero a la vez es el que exige mayor esfuerzo en su implementación (ver figura 3-1).

Existen tres tipos de conocimiento que pueden ser aprendidos, el *procedural*, el *declarativo* y el *cualitativo*.

El conocimiento procedural tiene que ver con la forma de realizar alguna tarea. Se aplica en casos como la solución de problemas de cálculo y generalmente se utiliza el formalismo basado en reglas de producción, en donde cada regla puede tomarse como una célula independiente de conocimiento que se explica al estudiante fuera de la estructura global del problema. Un ejemplo es el BUGGY [Brown y VanLehn, 1980], un sistema diseñado para mejorar las habilidades aritméticas.

Por su parte, el conocimiento declarativo se relaciona con dominios como la geografía, es decir, hechos organizados sobre los que es posible razonar. Lo que nos interesa en este caso es la enseñanza de hechos y principios. Un ejemplo de este tipo lo constituye el SCHOLAR [Carbonell, 1970], un STI para la enseñanza de la geografía sudamericana que emplea redes semánticas [Anderson, 1988].

Por otro lado, el conocimiento cualitativo se fundamenta en la simulación mental y el razonamiento sobre procesos dinámicos empleando estructuras causales. Se emplea para la



**Figura 3-1. Relación entre *efectividad pedagógica* y *esfuerzo de implementación***

enseñanza de habilidades para entender las estructuras causales de sistemas complejos, como los circuitos electrónicos. Actualmente se encuentra en una etapa de inmadurez comparado con otras áreas de la inteligencia artificial y no se ha logrado establecer una metodología de implementación apropiada. La figura 3-2 resume los aspectos más relevantes en la implementación del módulo experto.

1. **Modelos para la representación del conocimiento.**
  - a. *Caja negra*. Ofrece resultados correctos a los problemas, pero no puede dar explicación de sus conclusiones.
  - b. *Caja de cristal*. Almacena el conocimiento como los humanos, pero infiere en forma diferente.
  - c. *Modelo cognitivo*. Representa y usa el conocimiento en forma parecida a la manera en que lo hacen los humanos.
2. **Tipos de conocimiento.**
  - a. *Procedural*. Tiene que ver con la forma de realizar una tarea.
  - b. *Declarativo*. Se relaciona con dominios específicos.
  - c. *Cualitativo*. Se fundamenta en la simulación mental y el razonamiento sobre procesos dinámicos.

**Figura 3-2. Panorama del módulo experto.**

### **3.1.2. Módulo del estudiante.**

Un STI debe ser capaz de inferir el conocimiento del estudiante para adaptar su instrucción a las necesidades de éste. A esta acción de inferencia se le conoce como *diagnóstico del estudiante*, y al conocimiento inferido, resultado de ella es el *modelo del estudiante*. El modelo del estudiante es una estructura de datos que representa tanto lo que el estudiante sabe sobre la materia de interés como lo que desconoce de ésta. El diagnóstico del estudiante presenta muchas variantes dependiendo del tipo de conocimiento que se busca enseñar, la información disponible acerca del estudiante y el diseño global del sistema. Estas variantes y dificultades es lo que se da en llamar el *problema de modelación del estudiante*. Se han diseñado varios algoritmos para resolver el problema de modelación, y su aplicación está limitada a las características del sistema [VanLehn, 1988].

El modelo del estudiante tiene varios usos dentro del sistema, como son: el determinar cuál es el siguiente tópico a enseñar cuando el usuario ha dominado el tópico actual, cuándo y qué consejo ofrecer al estudiante cuando comete alguna equivocación sin poner en peligro el éxito pedagógico; generar nuevos problemas o preguntas de acuerdo al nivel de dominio que haya alcanzado; y basar la instrucción o las explicaciones en lo que ya conoce o domina el estudiante.

Para modelar el conocimiento del estudiante el sistema de diagnóstico necesita información proveniente del estudiante que dé indicios de su saber. El *ancho de banda* es un concepto que se emplea para identificar y categorizar la cantidad y calidad de la información de entrada al módulo de diagnóstico del estudiante. El mayor ancho de banda

son todos los *estados mentales* por los que pasa el estudiante cuando soluciona un problema, que es una categoría ideal. Obviamente no es posible que una máquina tenga acceso a esta información, lo que sugiere la búsqueda de una aproximación al nivel ideal en la cual se dispone de información que permite deducir los estados mentales más significativos. A este nivel se le llama *estados mentales aproximados*. Para tener acceso a los estados mentales pueden hacerse preguntas al estudiante o usar alguna otra técnica. Un sistema que usa un ancho de banda de estados aproximados es el LISP tutor [Reiser, Anderson, y Farrell, 1985], que tiene una interfaz basada en menús donde el estudiante define objetivos, estrategias y trozos de código para el diseño de programas en el lenguaje LISP. Por otro lado, hay ocasiones en donde sólo disponemos de algunos estados significativos, como es el caso de la solución de ecuaciones algebraicas. A éste se le conoce como ancho de banda de *estados intermedios*. Un ejemplo de ancho de banda de estados mentales intermedios es el sistema Spade [Miller, 1982], que actúa como un entrenador, observando en todo momento como el estudiante crea un programa, pero desconoce sus objetivos y estrategia. Una última categoría es aquella en la que sólo disponemos del estado final, es decir, el resultado de un problema o la respuesta a una pregunta. Este último es el llamado ancho de banda del *estado final*. El ejemplo clásico es el PROUST [Johnson y Soloway, 1984], que sólo tiene acceso al programa terminado en lenguaje PASCAL [VanLehn, 1988].

Otros aspectos importantes en la elección de la técnica de diagnóstico del estudiante, es el tipo de conocimiento y la técnica para determinar la diferencia entre el modelo del estudiante y el modelo del experto. Existen tres tipos de conocimiento, el *procedural plano* que es el más fácil de modelar e interpretar, el *procedural jerárquico*, cuya dificultad es intermedia, y el *declarativo*, que se distingue por ser el más difícil de modelar. El conocimiento procedural ha sido empleado para habilidades como la solución de ecuaciones algebraicas, juegos, aritmética y cálculo integral. La representación declarativa se ha utilizado en áreas como la geografía y la meteorología. Los límites de los tipos de representación no están claramente establecidos [VanLehn, 1988].

Las técnicas usadas para determinar la diferencia de conocimiento entre los modelos del estudiante y experto incluyen el *overlay*, en el cual el modelo del estudiante tiene su conocimiento representado en el mismo formato que el experto y es un subconjunto del modelo del experto. Una variante a la técnica del overlay es el uso de asignación de pesos al conocimiento aprendido por el estudiante según su grado de dominio. Otra técnica muy usada incluye un *catálogo de errores*, que contiene conceptos sin aprender y conceptos mal aprendidos. Existe también una técnica conocida como *catálogo de partes de errores*, en el cual los errores se ensamblan dinámicamente para acoplarse al comportamiento del estudiante. Los modelos del estudiante que hacen el diagnóstico más fácil son los modelos overlay con conocimiento procedural, donde los estados mentales están disponibles, y los que lo hacen más difícil son los modelos de bibliotecas de partes de errores sobre un conocimiento declarativo donde sólo se dispone del resultado final del razonamiento del estudiante. La figura 3-3 muestra las tres dimensiones de Kurt VanLehn [VanLehn, 1988], que permiten clasificar a las técnicas de modelación del estudiante.



1. **Ancho de banda.** ¿Cuánto de la actividad del estudiante está disponible al programa de diagnóstico?
  - a. *Estados mentales aproximados.* Toda la actividad, tanto física como mental está disponible.
  - b. *Estados intermedios.* Toda la actividad física observable está disponible.
  - c. *Estado final.* Sólo el estado final, la respuesta, está disponible.
  
2. **Tipo de conocimiento.** ¿Cuál es el tipo de conocimiento en el dominio de interés?
  - a. *Procedural plano.* Conocimiento procedural sin subobjetivos.
  - b. *Procedural jerárquico.* Conocimiento procedural con subobjetivos.
  - c. *Declarativo.* Conocimiento de un dominio específico.
  
3. **Diferencias entre el estudiante y el experto.** ¿Cómo difiere el modelo del estudiante del modelo del experto?
  - a. *Overlay.* Algunos temas del experto se desconocen.
  - b. *Catálogo de errores.* Además de los temas desconocidos, el estudiante puede tener conocimiento erróneo. Los errores se encuentran predefinidos en un catálogo.
  - c. *Catálogo de partes de errores.* Los errores se ensamblan dinámicamente para acoplarse al comportamiento del estudiante.

Figura 3-3. Las tres dimensiones del modelo del estudiante

### 3.1.3. Módulo instruccional.

Por otra parte, uno de los módulos esenciales en el diseño de los STI's es el módulo instruccional. Entre las funciones más importantes que desempeña se encuentra el control sobre la selección y secuencia del material presentado al estudiante, responder a preguntas planteadas por alumno y ayudar o aconsejar al estudiante cuando lo requiera. Aunque la función de este módulo es fundamental para el éxito del sistema como medio instruccional, poca atención se ha prestado a su inclusión dentro del STI.

De acuerdo al dominio, se han clasificado a los STI's en *tutores expositores* y *tutores procedurales*. Los *tutores expositores* conciernen a habilidades inferenciales con conocimiento real; ellos enseñan a los estudiantes un cuerpo de conocimiento real y las

habilidades necesarias para realizar inferencias sobre él. Por su parte, los *tutores procedurales* enseñan habilidades y procedimientos con aplicación fuera de las situaciones tutoriales, operan sobre la memoria y funcionan como lo hace un entrenador deportivo. Presentan ejemplos y exhiben habilidades de solución de problemas y sus ejercicios están orientados a la práctica [Halff, 1988].

Para avanzar hacia una selección de secuencia adecuada de conceptos es necesario tener presente los *conocimientos propedéuticos* del estudiante. Estos conocimientos son los necesarios para el aprendizaje del dominio del tutor inteligente, pero no para una ejecución y utilización experta del conocimiento. Así también es necesario considerar la formulación adecuada de una representación del conocimiento y la selección y secuencia de conceptos particulares a partir de esta representación que son los dos problemas centrales del *currículo*. El currículo podría conformar un concepto conocido como *red de enseñanza*, según D. Norman, basado en dos principios fundamentales. El primer principio establece que se debe dar prioridad a los conceptos que están más cercanamente relacionados al conocimiento existente, *principio de relación*. El segundo nos dice que es necesario discutir los conceptos generales antes que los específicos, *principio de generalidad*.

Para la selección de ejemplos y ejercicios se aconseja tener presente los aspectos de *manejabilidad, transparencia estructural, e individualización*. Por manejabilidad debemos entender que todo ejercicio podría ser resuelto, todo ejercicio debe ser comprensible a estudiantes que han avanzado a una parte de la curricula. R. Gagné y L. Briggs (1979) han recomendado analizar las habilidades a ser enseñadas dentro de un jerarquía de prerrequisitos, donde el nivel más alto es el objetivo primario. Por su parte, D. Merrill (1983) recomienda dos principios para la selección de ejemplos y ejercicios: el *principio de divergencia*, para la selección del ejemplo representativo, y el *principio de acoplamiento*, para la presentación de ejemplos positivos y negativos de lo que se está enseñando. La *transparencia estructural* permite que el estudiante induzca el procedimiento a partir de los ejemplos y ejercicios; cada ejercicio le dice algo al estudiante, cada ejercicio agrega un paso al procedimiento. Finalmente, la *individualización* establece que los ejercicios deben ser elegidos de acuerdo a las habilidades y debilidades que caracterizan al estudiante en el momento de la elección del ejemplo.

Entendemos por *instrucción* como la presentación del material al estudiante, las maneras de responder a las preguntas y las condiciones necesarias para realizar una intervención tutorial. Los tutores expositores utilizan el *diálogo* como método de presentación para lo cual pueden usarse los lineamientos derivados por Collins y Stevens (1982), quienes proponen que deben ser sensitivos al contexto tutorial. El diálogo que proponen Collins y Stevens es instructivo en el sentido de que su instrucción está condicionada por las respuestas del estudiante, pero estos diálogos se conforman de acuerdo a un plan rígido que no puede ser reformulado durante la interacción.

1. **Clasificación de los Sistemas de Tutorio Inteligente.**
  - a. *Tutores expositores.* Enseñan un cuerpo de conocimiento real y habilidades para usarlo.
  - b. *Tutores procedurales.* Enseñan habilidades y procedimientos con aplicación práctica.
2. **Aspectos para la selección de la secuencia adecuada de conceptos.**
  - a. *Principio de relación.* Dar prioridad a los conceptos que están más cercanamente relacionados al conocimiento existente.
  - b. *Principio de generalidad.* Discutir los conceptos generales antes que los específicos.
3. **Aspectos para la selección de ejemplos y ejercicios.**
  - a. *Manejabilidad.* Todo ejercicio debe ser comprensible para el estudiante.
  - b. *Transparencia estructural.* El estudiante debe poder inducir el procedimiento a partir de los ejemplos y ejercicios.
  - c. *Individualización.* Los ejercicios deben elegirse de acuerdo a las habilidades y debilidades del estudiante.

**Figura 3-4. Panorama para el diseño del módulo instruccional.**

Uno de los aspectos más importantes del tutorio inteligente es la oportunidad de interrumpir la instrucción para ofrecer un consejo o ayuda al estudiante cuando lo requiera, aun cuando no lo solicite. La intervención tutorial se requiere para mantener el control sobre la situación tutorial. Para la deducción de una intervención tutorial se han diseñado dos técnicas, el *trazado de modelo* y el *tutorio basado en temas*. En el trazado de modelo, se decide la intervención cuando el tutor observa que el estudiante se pierde o ya no es posible acoplar su comportamiento a una ruta de solución conocida. Por su parte, el tutorio basado en temas intenta identificar una ocasión propicia para la intervención; no restringe su intervención a la instrucción remedial y pueden ser más informativos, a diferencia del trazado de modelo [Half, 1988]. En la figura 3-4 se resumen los aspectos más importantes para la implementación del módulo instruccional.

### **3.1.4. Ambiente instruccional.**

Además, por otro lado, otros módulos que desempeñan una función sobresaliente son el ambiente instruccional y la interfaz de usuario. El ambiente instruccional se compone por

todos aquellos medios en que se apoya la instrucción constructiva del sistema, y pueden incluir situaciones y actividades de aprendizaje. El ambiente instruccional presenta al estudiante nueva información y experiencia para construir nuevo conocimiento [Burton, 1988].

El ambiente instruccional se compone de la parte del sistema que apoya las actividades que realiza el estudiante. Constituye en cierta medida un conjunto de herramientas disponibles al estudiante.

Nickerson (1986) define algunas características que debe tener el ambiente instruccional y su importancia desde el punto de vista educativo. La primera de estas características es el *constructivismo*, nos dice que el aprendizaje es la construcción de conocimiento nuevo sobre conocimientos existentes. El sistema debe permitir que el estudiante tome una actitud activa, y esté continuamente relacionando el nuevo conocimiento con el que ya posee. Es necesario identificar el conocimiento erróneo, preconcepciones, y corregirlo para poder construir sobre él de una manera efectiva.

Uno de los preceptos que enriquecen al ambiente instruccional es la adecuada selección de problemas o ejemplos para que el estudiante ensaye y aprenda con ellos. Por otro lado, una manera poderosa en que un ambiente puede apoyar la enseñanza es haciendo explícito un conocimiento implícito en su contenido. Un ejemplo de un sistema que hace consciente a los estudiantes de su proceso de inferencia es el BUGGY [R. Burton y J. Brown, 1982].

Una buena recomendación en el diseño de ambientes instruccionales es el presentar ideas y conocimientos desde múltiples perspectivas para permitir una mejor conceptualización del conocimiento.

Algunos de los sistemas de instrucción destacan por el diseño de su ambiente instruccional, entre ellos está el Lego Logo de S. Papert, que consiste en una colección de bloques de construcción, motores, interruptores y sensores para la construcción simulada de artefactos electromecánicos y la simulación y experimentación con su comportamiento. Otros ejemplos son el Envisioning Machine sobre el movimiento parabólico, el ARK de R. Smith, para experimentar y jugar con el movimiento de objetos que simulan el comportamiento de acuerdo a las leyes del movimiento [Burton, 1988].

### **3.1.5. Interfaz.**

Por su parte, la interfaz de usuario debe ser tal, que permita al estudiante un fácil aprendizaje del manejo del sistema, ya que de lo contrario éste gastará su energía intelectual en aprender su manejo en vez de emplearla en el aprendizaje del dominio de interés. El diseño de la interfaz es un problema de comunicación [Miller, 1988].

Una interfaz es el medio por el cual una aplicación se comunica con el usuario. Existen tres aspectos que deben considerarse al momento de especificar su diseño, estos son, los *factores humanos*, la *presentación* y la *interacción*. Es importante mencionar que la interfaz es el módulo del sistema que por lo general influye en una mayor proporción en la aceptación del sistema por parte del usuario.

El aspecto más importante a considerar de los tres mencionados, son los *factores humanos*. Es esencial identificar y entender la capacidad del usuario, sus conocimientos, experiencia, habilidades físicas, percepción y características del espacio donde se usará el sistema. Dos principios fundamentales en este aspecto son: *ceder el poder al usuario* y *reducir la carga de información* sobre el usuario. El primer principio establece la necesidad de darle al usuario el control en lugar de que se sienta manejado por el sistema. Lo anterior significa que el usuario puede controlar el tipo de información presentada, la velocidad de presentación y la manera en la cual es presentada. Además, una buena interfaz evita sobrecargar la memoria del usuario ofreciendo la lista de comandos y opciones que puede ejecutar, presentando sólo datos relevantes y organizando la información de una manera significativa.

Por otra parte, cuando hablamos de *presentación* nos referimos al ámbito visual de la interfaz. Dentro de la presentación caen el color, las formas, los tipos de letra, los tamaños de cada objeto y los espacios vacíos. Entre los principios recomendados al diseñar la presentación se encuentran la *estética*, la *representación significativa* de los objetos y la *consistencia* de la interfaz.

Finalmente, la *interacción* es el medio por el cual el usuario controla la ejecución de una aplicación. El concepto de señalar-seleccionar, representa la esencia de la interacción. La interacción puede realizarse por el teclado o mediante algún dispositivo tal como el ratón. Los principios más importantes de este aspecto son, permitir la *manipulación directa de objetos* (iconos o símbolos), ofrecer *retroalimentación inmediata* y *tolerancia a los errores del usuario*.

A todo lo dicho podemos agregar que la interfaz de usuario es un conjunto de posibles *eventos de interacción*. Entendiendo como evento las regiones espacio-temporales, incluidos los cambios de estado. Un evento de interacción es aquel evento que se suscita a causa de una acción del usuario o de la aplicación, es decir, del usuario a la aplicación o de la aplicación al usuario.

Un concepto que juega un rol muy importante en el diseño de la interfaz es *el modelo del usuario* acerca de la aplicación. El modelo del usuario apoya al usuario planificando sus actividades y decidiendo la próxima tarea a realizar. Es decir, el modelo del usuario hace sensible lo que el usuario siente y hace [Marcus, 1995]. La figura 3-5 muestra los aspectos para especificar el diseño de la interfaz.

1. **Factores humanos.** Identificar y entender la capacidad del usuario, sus conocimientos, experiencias y habilidades físicas.
  - a. *Ceder el poder al usuario.* Dar al usuario el control sobre el sistema.
  - b. *Reducir la carga de información.* Ofrecer lista de comandos, opciones, datos relevantes y organización de información.
2. **Presentación.** Color, formas, tipos de letra y tamaños de objetos.
  - a. *Estética.*
  - b. *Representación significativa de los objetos.*
  - c. *Consistencia.*
3. **Interacción.** Medio por el cual el usuario controla la ejecución de una aplicación.
  - a. *Ofrecer retroalimentación inmediata.*
  - b. *Tolerancia a los errores del usuario en el uso del sistema.*

Figura 3-5. Aspectos para especificar el diseño de la interfaz.

## 3.2. Aplicaciones similares.

### 3.2.1. El Practical Algebra Tutor (PAT).

El sistema PAT está siendo desarrollado por el grupo ACT (Advanced Computer Tutoring) dirigido por John Anderson en la Universidad de Carnegie Mellon, quienes se proponen la creación de sistemas de tutorio inteligentes basados sobre una arquitectura cognitiva que apoye la enseñanza y el aprendizaje. Todos sus desarrollos están sustentados en una teoría cognitiva conocida como *teoría cognitiva ACT*. Esta teoría propone que el ser humano resuelve problemas basado en un conjunto de reglas de producción, y describe además cómo son aprendidas y ejecutadas estas reglas durante el proceso de solución de problemas.

Este sistema tiene como objetivo desarrollar en el estudiante habilidades algebraicas que sean aplicables en situaciones de la vida real, tales como elegir la compañía de teléfonos que ofrece un servicio más económico, o determinar el consumo de energía

eléctrica esperado para los próximos tres meses, o bien, determinar bajo qué condiciones es mejor rentar los servicios de cada una de las dos compañías que ofrecen un mismo servicio.

Además, hace énfasis en el empleo de varios tipos de representación de datos e información. La información puede ser representada en forma escrita (palabras), algebraica (simbólica), gráfica y discreta (tabular). De esta forma el estudiante va construyendo su propia solución al problema planteado, y adquiere habilidades para representar información de diferentes maneras.

El sistema inicia con una descripción escrita de una situación y una serie de preguntas relativas que se espera que el estudiante responda (ver figura 3-6).

<b>PROBLEM STATEMENT</b>
<b>GENERAL DIRECTIONS</b>
Our goal is to complete the worksheet and to construct a graph showing two different algebraic equations on the same set of axis. More directions about worksheet entry and graph construction are available under the PROBLEM menu.
<b>PROBLEM</b>
We need to rent a large truck for one week. Rental companies charge on initial cost plus on additional cost for each mile driven. One company, Hertz, will rent a 27 foot truck to us for \$325 plus \$0.21 per mile, will rent us the same size truck for \$585 plus \$0.13 per mile.
<ol style="list-style-type: none"><li>1. What is the initial cost, in dollars, of renting this model truck from Hertz? Form Avis?</li><li>2. How much would it cost to rent the truck from Hertz if we will be driving it 500 miles? From Avis?</li><li>3. How many miles we need to drive the truck so that the cost of rental from each company would be equal?</li></ol>
At any time, you may define a variable for the number of miles driven and use this variable to write rules for the cost of renting the truck from Hertz and the cost of renting the truck from the Avis.

**Figura 3-6. Pantalla del sistema PAT que presenta una descripción escrita de una situación (tomada de ACT Research Group, 1997).**

El estudiante representa de varias maneras la información contenida en la descripción escrita, tales como gráficas o tablas (ver figura 3-7 y 3-8), y basándose en ellas intenta responder a los cuestionamientos hechos por el sistema. Para poder solucionar cada pregunta el estudiante debe manejar efectivamente las herramientas de representación de información. En todo momento el estudiante es acompañado por el sistema ofreciéndole

ayuda siempre que sea necesario, o proponiendo nuevas alternativas de solución. Durante todo el proceso el sistema va identificando las áreas de dificultad para el estudiante.

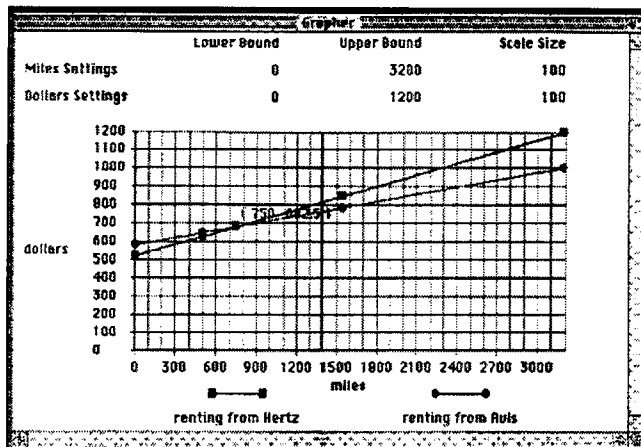


Figura 3-7. Representación gráfica del problema usando el sistema PAT (tomada de ACT Research Group, 1997).

Worksheet			
	miles driven	cost of renting from Hertz	cost of renting from Avis
UNIT	miles	dollars	dollars
FORMULA	$x$	$0.21x + 525$	$0.13x + 595$
1	0	525	595
2	500	630	650
3	1547.619	850	786.1905...
4	3192.3077...	1195.3846...	1000
5	750	682.5	682.5

Figura 3-8. Representación tabular del problema usando el sistema PAT (tomada de ACT Research Group, 1997).

En cada lección el sistema procede aumentando el grado de dificultad progresivamente. En las primeras lecciones busca que estudiante aprenda a identificar las cantidades importantes y aprenda a representar información en forma gráfica y tabular; en las lecciones intermedias el estudiante aprende a representar información en forma simbólica, y a resolver fórmulas lineales y problemas donde se pone énfasis en la utilización de gráficas para hacer proyecciones y encontrar intersecciones; en las lecciones finales el estudiante se ocupa de la solución de ecuaciones con números grandes y negativos, sistemas de ecuaciones, porcentaje y razonamiento proporcional.

000309



El sistema implementa un enfoque de tutorio llamado *trazado de modelo* (model-tracing). La premisa básica de este enfoque de tutorio es desarrollar un modelo cognitivo para interpretar el comportamiento del estudiante en la solución de problemas y guiarlo durante su estudio. Está representado como un conjunto de reglas de producción. El modelo cognitivo representa el comportamiento “ideal” del estudiante y debe ser capaz de producir toda ruta de solución aceptable del problema.

## **Capítulo 4. Diseño General del Sistema.**

El presente capítulo presenta el diseño global del sistema de tutorío propuesto en este trabajo de tesis. Está organizado en dos secciones principales, la primera describe el diseño funcional del sistema, las principales operaciones que puede realizar y se presentan los bloques principales y su interacción. La segunda sección presenta la metodología seguida para el diseño de la interfaz de usuario, un estudio sobre los usuarios potenciales y el diagrama de navegación del sistema. Sin embargo, los aspectos relativos a las técnicas de inteligencia artificial no son cubiertos en este capítulo, estos se describen posteriormente en el capítulo cinco.

### **4.1. Diseño Funcional.**

A continuación se presenta una descripción del diseño del sistema de tutorío inteligente propuesto. En esta sección se describen los aspectos más relevantes del diseño funcional, prerrequisitos y necesidades y actividades de aprendizaje.

#### **4.1.1. Prerrequisitos y necesidades.**

El sistema de tutorío inteligente cuyo diseño describimos a continuación tiene como propósito la enseñanza de la simplificación algebraica mediante el uso de las Leyes de los Exponentes y, está dirigido a alumnos de nivel preparatoria. Los prerrequisitos para el aprovechamiento eficiente de su uso es que el alumno domine las matemáticas de nivel secundaria; específicamente que domine la aritmética y que sepa lo que es una expresión algebraica, es decir, entender el significado de los conceptos de variable, constante y exponente y, la función de cada uno de estos conceptos dentro de una expresión algebraica, y la explicación teórica de las Leyes de los Exponentes. Igualmente es recomendable que haya manejado programas computacionales con ambiente gráfico.

Si identificamos los conocimientos actuales del alumno recién egresado de secundaria como el estado actual y como estado final deseado los conocimientos adquiridos por el

alumno después de usar el sistema, podemos definir las necesidades como la diferencia entre estos dos estados. Es decir, las necesidades son todos los conceptos y procedimientos que aprenderá el estudiante para alcanzar el estado deseado. Las necesidades que pretende cubrir el presente sistema es la aplicación de las Leyes de los Exponentes en la simplificación de expresiones algebraicas. Con el uso de este sistema el estudiante tendrá acceso a los conceptos relativos a las Leyes de los Exponentes, como, exponente cero, exponente uno, enunciado de las Leyes de los Exponentes.

#### 4.1.2. Especificación funcional.

De acuerdo a las metas de diseño descritas en la sección 2.3.1, el sistema debe, por un lado, ofrecer al estudiante un medio para la práctica de ejercicios de simplificación de expresiones y recibir retroalimentación de acuerdo a su desempeño, y además, funcionar como herramienta del maestro para el diagnóstico del estudiante. En base a estas metas de diseño, el sistema requiere un módulo para la práctica de ejercicios. Igualmente es necesario que el sistema tenga una base de datos que almacene el diagnóstico del desempeño de cada estudiante y los datos generales de cada usuario, y además, que esta base de datos pueda ser consultada por el maestro y el estudiante. Dadas las características de diseño descritas arriba, es preciso que se identifique a cada usuario para que el sistema determine que operaciones puede realizar mientras usa el sistema y para utilizar y actualizar la información del usuario. En la figura 4-1 puede apreciarse un diagrama de bloques que describe la funcionalidad global del sistema propuesto.

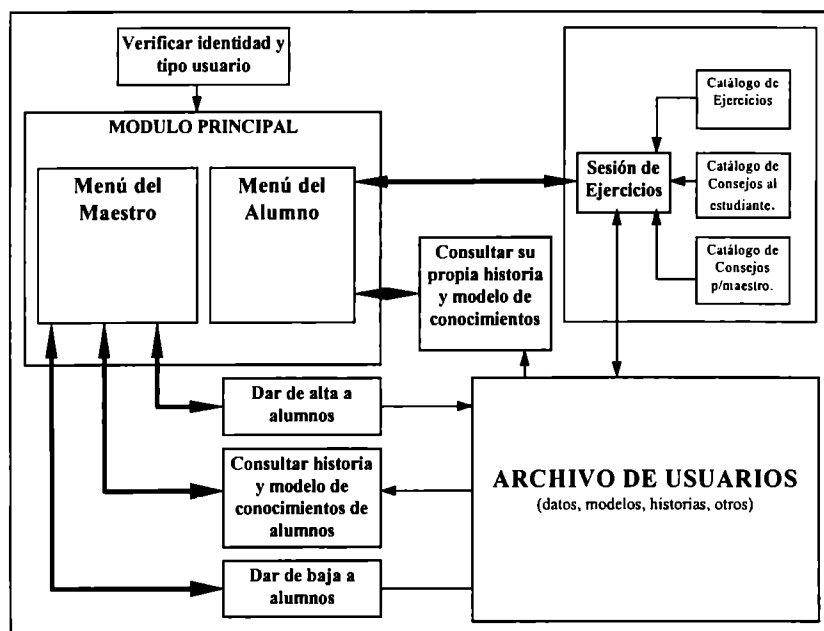


Figura 4-1. Diagrama de bloques que describe la funcionalidad global del sistema.

## **Módulo para verificar identidad y tipo de usuario.**

Para lograr un control del desempeño de los estudiantes que usan el sistema, se crean una serie de archivos por cada estudiante; por lo tanto es preciso identificar al usuario cada vez que se ejecute el sistema para tener acceso a toda la información relacionada con él. En este bloque el sistema pregunta al usuario si es un alumno o un maestro. Posteriormente solicita el nombre del usuario, con el propósito de buscar en la base de datos la información referente a ese usuario (historia de un alumno, grupo de alumnos a cargo de un maestro, etc.) y modificarla de acuerdo a las acciones que solicite o realice. Los alumnos que estén dados de alta tendrán un archivo que almacena sus datos y conocimientos. De cualquier modo, no sólo los alumnos que estén dados de alta pueden usar el sistema. Un alumno no registrado puede usar el sistema, pero no se guardará la *historia* de su evolución. Por *historia* entendemos el registro de todos los errores que ha realizado el estudiante.

## **Módulo principal del maestro.**

En esta pantalla el maestro tiene la opción de:

### a.- Dar de alta a un alumno.

Los maestros puede dar de alta o baja a alumnos en del sistema. El sistema sólo guarda la historia de los alumnos registrados, de esta manera, cada vez que el alumno use el sistema no tiene que resolver de nuevo ejercicios de un nivel inferior a su capacidad. Además, el maestro puede revisar su evolución, desempeño, sus dificultades y habilidades desarrolladas, abriendo un archivo del alumno, donde se describe lo que hemos denominado *historia* del alumno.

Cada maestro es responsable de dar de alta y baja a los alumno bajo su cargo. El propósito es mantener sólo datos útiles almacenados en los archivos de datos del sistema y evitar el crecimiento desmedido.

### b.- Dar de baja a un alumno.

El maestro puede dar de baja a los alumnos bajo su cargo cuando considere que es necesario. La ejecución de este comando borra toda la información sobre ese alumno en el sistema.

### d.- Consultar la *historia de errores y modelo de conocimientos* del alumno.

El maestro puede consultar la información sobre las habilidades de los alumnos que haya dado de alta, ya que el sistema lleva un registro de los errores que el estudiante comete y de su modelo de conocimientos. El maestro puede tomar nota del estado de las habilidades del estudiante y de su desempeño durante la solución de ejercicios.

## **Módulo principal del alumno.**

En esta pantalla el alumno tiene la opción de:

### a.- Iniciar sesión de ejercicios.

En esta sesión se ponen a prueba las habilidades de los alumnos para la solución de expresiones algebraicas. Básicamente consiste en presentarle al estudiante una expresión algebraica para que la resuelva y observar su desempeño y, hacer anotaciones en su archivo de historia. Esta etapa se describe más ampliamente en la sección de *actividades de aprendizaje* de este capítulo.

### b.- Consultar su propia *historia de errores y modelo de conocimientos*.

El alumno puede consultar su propia historia de errores y su modelo de conocimientos para permitirle conocer y corregir sus errores en la solución de expresiones algebraicas y observar la manera en que van evolucionando sus conocimientos.

## **4.1.3. Actividades de aprendizaje.**

La principal actividad de aprendizaje se presenta en la *sesión de ejercicios* del sistema. En la figura 4-2 puede apreciarse un diagrama de flujo de la sesión de ejercicios. A continuación se describe esta sesión:

### **Sesión de ejercicios.**

#### a. Se determinan las características del próximo ejercicio a seleccionar en base *al modelo del estudiante* (sección 3.1.2).

El modelo del estudiante consiste en una representación de las leyes de los exponentes y del álgebra que el alumno domina, las leyes que no domina, y las leyes mal aplicadas (llamadas incorrectas). La revisión le permite al sistema seleccionar un ejercicio apropiado en el siguiente paso.

#### b. Se genera un ejercicio de acuerdo a las características obtenidas en el paso anterior. Este módulo toma de un *catálogo de ejercicios* una estructura base y a partir de esa estructura se genera aleatoriamente el ejercicio.

La selección de la estructura base y la generación del ejercicio se realiza de acuerdo a los principios (leyes de los exponentes u otras leyes) requeridos para solucionarlo, números de variables y complejidad de la expresión. Todas las características del nuevo ejercicio son tomadas del diagnóstico o modelo del estudiante.

- c. Se presenta el ejercicio al estudiante.
- d. El estudiante realiza paso a paso la solución del ejercicio seleccionando de las opciones que le presente el sistema.

El sistema le presenta al estudiante un conjunto de opciones (normalmente de tres a cinco), entre las que el estudiante debe seleccionar la que considere más apropiada. El hecho de seleccionar un expresión ya escrita evita errores de edición de la expresión por parte del estudiante y le permite al sistema un proceso de diagnóstico del estudiante más sencillo. En cualquier momento el estudiante tiene la opción de terminar la sesión de ejercicios, abandonar el ejercicio actual o dar por terminado el ejercicio.

- e. Al terminar de resolver el ejercicio el sistema actualiza el *modelo del estudiante* y se archiva la actualización.

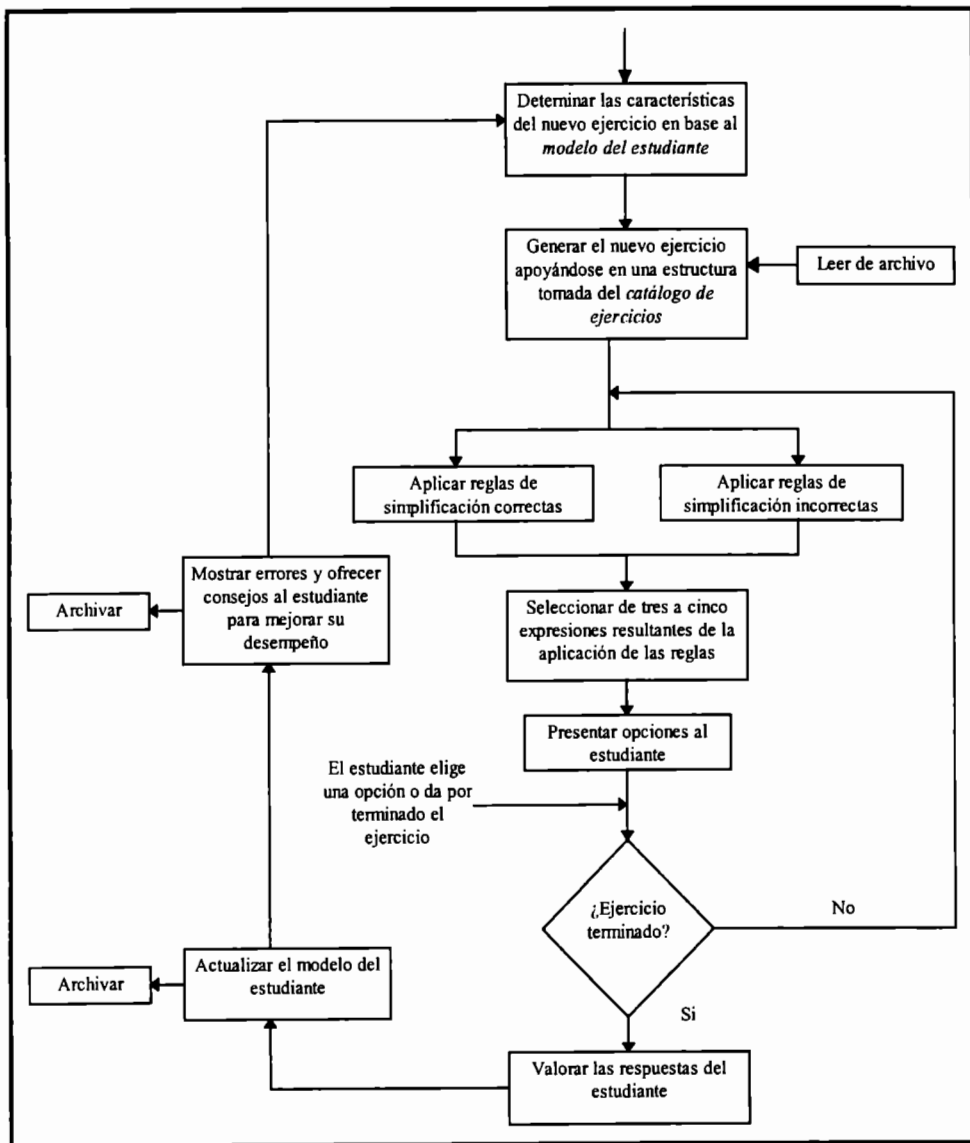


Figura 4-2. Diagrama de flujo de la *sesión de ejercicios*.

- f. Si el sistema detectó errores durante la solución del ejercicio, ofrecerá consejos apropiados al estudiante. El consejo puede consistir en indicarle el error al estudiante, recordarle alguna ley algebraica que haya olvidado.
- g. Se reinicia el ciclo en el inciso *a*.

## **4.2. Diseño de la interfaz.**

### **4.2.1. Metodología de diseño.**

Para el diseño de la interfaz de usuario de un sistema no existen una metodología que garantice el éxito, pues es en gran medida es una actividad creativa en sus etapas críticas. Existen, sin embargo, diferentes enfoques para diseñar una interfaz de usuario, como son el enfoque basado en modelos descrito por Janlert [Janlert, 1989], o el enfoque de la creación de metáforas reconocibles por el usuario [Hagen, 1995]. En lo que si se está de acuerdo entre los investigadores es que es necesario un profundo estudio del usuario potencial para lograr un diseño que sea aceptable por el usuario. En base a lo anterior, para la creación de esta primer propuesta de interfaz de usuario para el STI, se realizaron las siguientes etapas de diseño:

- a) **Definición del usuario.**  
Se determina quien o quienes son los usuarios potenciales del sistema.
- b) **Modelo del diseñador acerca del usuario.**  
Se realiza una descripción del conocimiento que el diseñador tiene sobre los usuarios potenciales en aspectos como habilidades físicas, habilidades cognitivas, aspectos culturales, entre otros.
- c) **Modelo del diseñador acerca de la actividad.**  
Se realiza una descripción acerca de la forma en que el diseñador entiende la actividad que va a realizarse con el sistema.
- d) **Estudio cualitativo del usuario: encuesta a usuarios potenciales.**  
Los dos puntos anteriores se refinan mediante la aplicación de una encuesta a los usuarios portenciales. Esta encuesta se encuentra en el anexo de esta tesis.
- e) **Definición de metáforas.**  
De acuerdo a los resultados de la encuesta se definen las metáforas con que trabajará el sistema.
- f) **Diseño de interacción.**  
Se crea el diseño de interacción más apropiado a los propósitos del sistema.
- g) **Creación de las pantallas principales.**  
Se crean las diferentes pantallas que contendrá el sistema. Las pantallas creadas se encuentran en el anexo de esta tesis.

#### 4.2.2. Modelo del diseñador acerca de la actividad.

El *modelo del diseñador acerca de la actividad*, es el punto de vista que tiene el diseñador de la actividad: objetivo, conceptos, métodos, instrumentos, patrones de trabajo, etc. El modelo de la actividad incluye la situación en la cual la actividad se origina y se resuelve. Es importante analizar este modelo porque generalmente el usuario ya trae un modelo preconcebido de la forma en que se realiza la actividad que propone el sistema, o bien espera del sistema características encontradas en aplicaciones similares o provenientes de sus experiencias previas en esa actividad [Janlert, 1989]. En los párrafos siguientes de esta sección se describe el modelo del diseñador acerca de la actividad.

La mayor parte de los maestros de matemáticas perciben la enseñanza como un arte en el que combinan varias disciplinas y habilidades, como psicología, comunicación, intuición, creatividad, etc., además de su área de conocimiento. En cuanto a la actividad de enseñar, desean conocer qué errores comete el estudiante y cuál es la razón por la que cometen esos errores. Su paradigma de trabajo es proponer ejercicios al alumno y observar los pasos que realizan en su solución y los errores que comete, interpreta los errores relacionándolo con una causa probable (hipótesis), propone nuevos ejercicios para probar las causas hipotéticas de los errores o para proponer una nueva hipótesis.

La manera en que normalmente trabaja el estudiante es sobre un cuaderno o un pizarrón; y la práctica de ejercicios dentro del aula se da aproximadamente de la siguiente forma:

- a. el maestro le muestra al estudiante las reglas y los aspectos teóricos necesarios para resolver los ejercicios,
- b. el maestro resuelve ejercicios para ejemplificar el uso de las reglas analizadas y,
- c. le propone ejercicios al estudiante para que los resuelva y reconocer la existencia de errores en el conocimiento adquirido por el estudiante, relacionándolos con alguna causa probable.
- d. de acuerdo a las respuestas del estudiante se le proponen nuevos ejercicios cada vez de mayor complejidad, o para probar las causas probables de error.

El maestro prefiere el autoestudio por parte del estudiante en la solución de ejercicios, antes que llevarlo de la mano, sin embargo, el alumno prefiere en muchos casos ser llevado de la mano o, que el maestro le diga la solución del ejercicio; y por lo regular tardan en tomar confianza al aprender una nueva regla de simplificación de expresiones algebraicas.



### 4.2.3. Modelo del diseñador acerca del usuario potencial.

Cuando hablamos del *modelo del diseñador acerca del usuario*, nos referimos al conocimiento que tiene el diseñador del sistema acerca del usuario potencial. Este conocimiento va desde el conocimiento del sentido común de los seres humanos, hasta el conocimiento específico relativo a la interacción entre el humano y la computadora. El modelo del diseñador acerca del usuario debe ser específico a un usuario o grupo de usuarios. Lo más importante que debe conocer el diseñador es el rol de usuario en relación a la aplicación y la actividad [Janlert, 1989]. Los siguientes párrafos muestran el análisis de los usuarios potenciales que cubre aspectos como: *habilidades físicas, habilidades cognitivas, aspectos culturales, área de trabajo y personalidad* [Campbell, 1997].

#### Definición del usuario.

Alumnos de preparatoria del ITESM del tercer semestre y maestros de matemáticas de apoyo académico de las preparatorias del ITESM.

#### Alumnos:

**Habilidades físicas.** Practican deporte en forma continua, por lo que debe esperarse que tengan una gran habilidad física, en otras palabras, que sean muy exactos en sus movimientos y con un excelente pulso. Puede esperarse que gozen de muy buena salud y todos sus sentidos estén trabajando a un nivel óptimo, por esta razón es posible afirmar que tienen una amplia capacidad para identificar objetos y colores en el contexto y de reaccionar ante estímulos visuales y auditivos, que reconocen fácilmente sonidos y, que tienen facilidad para leer textos de tamaño pequeño.

Los alumnos de preparatoria de ITESM pertenecen a una clase privilegiada, por lo que en caso de deficiencias visuales pueden adquirir lentes o recibir algún tratamiento especial. Se tiene la posibilidad de que existan alumnos con problemas de daltonismo, aun cuando no se detectó ninguno en una encuesta realizada a una muestra de estos alumnos, pero dado que es posible su ocurrencia en algún alumno, debe considerarse este aspecto en la selección de colores para interfaz.

**Habilidades cognitivas.** No son muy pacientes ni rápidos para la lectura. Por lo general están familiarizados con el equipo de cómputo y el ambiente gráfico. Se espera que el alumno conozca la aritmética y tenga algunos conocimientos teóricos sobre álgebra y la simplificación de expresiones algebraicas.

Algunos símbolos con significado especial para alumnos y maestros:

1. *Signo igual* : Representa la igualdad de dos expresiones. Podría usarse para que el alumno aplique una regla sobre la expresión actual.
2. *Libreta* : Representa el lugar para realizar ejercicios y hacer notas. Podría usarse una libreta abierta para iniciar una sesión de ejercicios.

3. *Lápiz* : Representa agregar o proponer algo nuevo durante la solución de un ejercicio.
4. *Borrador* : Representa la acción de borrar para corregir.
5. *Libro* o *Maestro* : Representa el objeto o persona que aclara dudas.

**Aspectos Culturales.** Los estudiantes de nivel preparatoria tienen una edad que oscila entre los 14 y 18 años. Por lo general desconocen el significado de muchas palabras del español no usadas en su ambiente. Les gusta divertirse, y todo lo que les divierte les despierta la atención. Rechazan las actitudes demasiado serias, buscan la jovialidad, la charla fácil, viven el presente.

**Area de trabajo.** El lugar donde se usará el sistema tiene refrigeración y calefacción adecuadas, con óptima iluminación, ruido continuo medio y mobiliario cómodo. Supongo que los usuarios usarán el sistema para estudiar el tema de interés de media a una hora diaria durante dos semanas como máximo.

**Personalidad.** Por lo general son extrovertidos, muy sensibles, perceptivos y orientan su atención más hacia sus sentidos que hacia su pensamiento.

### Maestros:

**Habilidades físicas.** Realizan poco ejercicio y que por lo tanto son de habilidades físicas medias, tanto en la exactitud y destreza manual como en la percepción visual. Presentan más dificultades que un estudiante para distinguir textos e identificar objetos dentro de algún contexto.

**Habilidades cognitivas.** En general tienen un tiempo corto para el aprendizaje, tienen una amplia capacidad de retención y son rápidos en la escritura y la lectura; tienen estudios de profesional y maestría y, están familiarizados con el uso de equipo de cómputo y ambiente gráfico.

**Aspectos Culturales.** La edad de los maestros oscila entre los 26 y 40 años, dominan el español y el inglés y son de nacionalidad mexicana.

**Area de trabajo.** Trabajan en lugares bien iluminados, temperatura agradable, libre de vibración, con un nivel medio de ruido y con mobiliario cómodo. Pasan en promedio ocho horas en lugar de trabajo y usarán el sistema entre media y una hora diaria durante tres semanas.

**Personalidad.** Son extrovertidos, intuitivos, juiciosos y orientados a actuar guiados por el pensamiento.

#### 4.2.4. Diseño de Interacción.

##### a. Perspectiva a la que se orienta el diseño propuesto.

Se habla de tres perspectivas en las que se puede basar el diseño. Primeramente tenemos a la perspectiva *orientada a la actividad*, que consiste en que la interfaz le permite al usuario concentrarse en desarrollar el trabajo o actividad que él desea, descargando su memoria de exceso de comandos o procedimientos complicados. El usuario conoce las entradas que debe proporcionar al sistema a través de la interfaz y sabe las salidas que le proporcionará. Otro tipo de perspectiva es la de *flexibilidad de uso*, esta perspectiva le permite al usuario conocer más acerca del sistema, sus capacidades y forma de funcionamiento. En esta perspectiva, además de conocer las entradas y salidas del sistema, el usuario sabe los procesos que se llevan a cabo dentro de él al ejecutar las operaciones. Un tercer tipo de perspectiva, es la conocida como *orientada al mecanismo*, esta perspectiva le permite al usuario conocer perfectamente las entradas y salidas, su funcionalidad, capacidades y limitaciones del sistema. El usuario debe conocer profundamente el sistema, lo que le permite corregir errores y hacer mejoras e innovaciones. El usuario tiene a su alcance el control completo sobre el sistema [Gentner, 1996].

El diseño del sistema que se propone en este artículo está *orientado a la actividad*. Es muy importante implementar esta perspectiva, pues así el estudiante no se distrae y se concentra más en la solución de problemas, que es la actividad más importante del sistema. El sistema estará compuesto por sólo las opciones mínimas que permitan el desarrollo de la actividad propuesta, pero a la vez, todas las opciones necesarias que den al estudiante el control sobre lo que realiza el sistema.

##### b. Nivel de interacción que implementará el diseño propuesto.

Cuando hablamos de niveles de interacción de un sistema, nos referimos al hecho de ubicar el sistema en el nivel de comunicación que se alcanza entre el sistema y el usuario por las características de la interfaz. Se describen tres niveles de interacción que puede implementar el diseño, estos son: *físico*, *funcional* y *racional*. Tenemos primero al *físico*, en este nivel, el usuario manipula la interfaz de un sistema computacional sin saber qué hacen los iconos, opciones y comandos disponibles, ni para qué sirven cada uno de ellos, sin embargo, se logra la interacción. Un segundo nivel es el *funcional*, en este nivel, el usuario conoce las entradas y salidas del sistema, sin embargo, no sabe lo que sucede internamente. El usuario interactúa con el sistema con la seguridad de que si oprime cierta tecla o botón, el sistema responderá de cierta manera. Es una interacción en términos de símbolos, en forma metafórica, la interacción se presenta como un “diálogo”. El tercer y último nivel de interacción es el *racional*, este nivel agrega pragmatismo; y con el pragmatismo se alcanza un diálogo real. No únicamente para la cooperación sino también para la argumentación y la crítica. En este nivel el usuario alcanza un profundo conocimiento del sistema, obteniendo de él un uso óptimo [Janlert, 1989].

Se espera lograr implementar un *nivel de interacción funcional*, y que el nivel físico sea superado lo más rápido posible en el caso de los usuarios novatos. Para un sistema educativo no es apropiado que el usuario se entretenga demasiado tiempo en el nivel físico pues se engañaría el sistema al querer conocer los conocimientos del estudiante. Por otro lado, es innecesario para los propósitos del sistema que el estudiante llegue a un nivel racional.

### c. Estilo de diálogo que implementará el diseño propuesto.

El estilo de diálogo son los medios que el sistema y el usuario emplean para comunicarse e interactuar. Entre los principales estilos de diálogo tenemos al estilo *orientado a comandos*, que es en el cual el usuario manipula el sistema por una serie de órdenes o comandos oscuros y difíciles de recordar, como los usados por el sistema UNIX. Otro tipo de estilo de diálogo es la *manipulación directa*, que es el estilo con el cual el usuario puede interactuar con el sistema con ayuda de iconos, símbolos o índices; entre sus virtudes está que el usuario no tiene por qué recordar tanta información para interactuar con el sistema y porque es una técnica cercana a la manera como el usuario interactúa con los objetos reales. Otro ejemplo es el *lenguaje natural*, en esta forma de diálogo, la interacción con el sistema puede ser en forma escrita (texto) o en forma hablada (voz).

Se busca la implementación en ambiente de *manipulación directa*, apoyado por iconos, símbolos e índices. La razón de esto es que este estilo de diálogo ha demostrado mayor efectividad en el diseño de sistemas educativos, y por que es más fácil la identificación de las diferentes metáforas, iconos y símbolos que se representan en la interfaz, brindándole al usuario un ambiente agradable y funcional.

## 4.2.5. Metáforas que propone el diseño.

La *metáfora* es un artificio literario que consiste en el hecho de describir algo apoyándose en otra cosa que puede ser mejor conocida, más fácil de entender o bien que le da un toque agradable o de belleza a lo que se pretende describir. En el diseño de sistemas interactivos, la metáfora es la representación de algo en la interactividad de un sistema apoyándose en algún concepto o actividad común para el usuario que le permita un fácil y rápido aprendizaje de su manejo. Uno de los desafíos importantes para el futuro del diseño de Interfaces de Usuario (UI) es el diseño de metáforas, porque la comunicación entre los seres humanos normalmente está llena de referencias metafóricas. Las metáforas ayudan a la gente a entender y recordar las cosas. Una buena metáfora hace más fácil de aprender, entender y usar un sistema [Hagen, 1995].

El diseño propuesto implementa un conjunto de metáforas que mutuamente se apoyan en la descripción de los escenarios de desarrollo de las actividades, principalmente del estudiante, ya que es el usuario más importante del sistema. Como metáfora principal, el diseño propone el *salón de clases*, y se concretará con el diseño de una interfaz de usuario

que represente el ambiente escolar. Para apoyar esta metáfora se unen otras metáforas de importancia, ellas son, el *maestro*, que representa la fuente de conocimiento, la *libreta*, que los estudiantes ven como el objeto donde es posible la práctica de ejercicios, el *borrador*, que el estudiante lo visualiza como un objeto que le permite eliminar alguna acción realizada, y el *lápiz*, que le permite proponer alguna solución a un ejercicio. Con estas metáforas se pueden crear los principales ambientes del sistema, especialmente en la creación de la pantalla de la sesión de ejercicios que podemos ubicarla en el diagrama de navegación que se describe en la siguiente sesión.

#### 4.2.6. Diagrama de Navegación del Diseño propuesto.

La figura 4-3 contiene el diagrama de navegación del diseño propuesto. La información relativa a la funcionalidad de cada pantalla puede encontrarse en la sección de *Especificación funcional* de este capítulo.

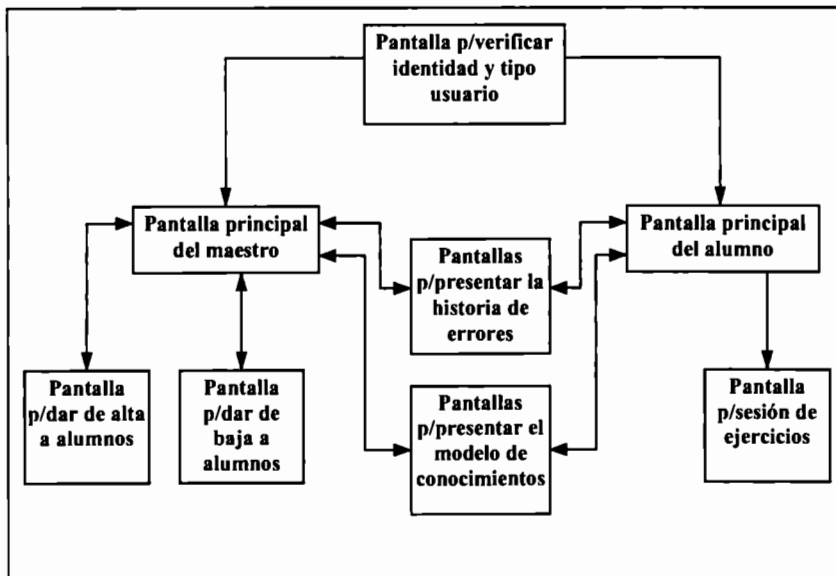


Figura 4-3. Diagrama de navegación.

#### 4.2.7. Diseño de las pantallas del sistema.

El diseño de las pantallas que conforman el sistema se realizó considerando todos los aspectos estudiados en las diferentes etapas del proceso. El conjunto de pantallas creadas se encuentran en el anexo de este escrito. Los aspectos de mayor relevancia en su diseño fueron:

- 1.- empleo de metáforas, símbolos y ambientes propios del salón de clases,
- 2.- que pueden ser implementadas para trabajar bajo un ambiente gráfico, y
- 3.- permitir una forma de interacción de manipulación directa por parte del usuario.

## Capítulo 5. Diseño de los módulos inteligentes.

Habiendo analizado el diseño general del sistema en el capítulo cuatro, en este capítulo se describirá cada uno de los componentes inteligentes del sistema: el *módulo experto*, el *módulo del estudiante* y el *módulo instruccional*. Se presenta una descripción de la estructura que soporta el sistema completo (ver figura 5-1) y se centra esencialmente en el módulo de la sesión de ejercicios del sistema descrito en el capítulo cuatro.

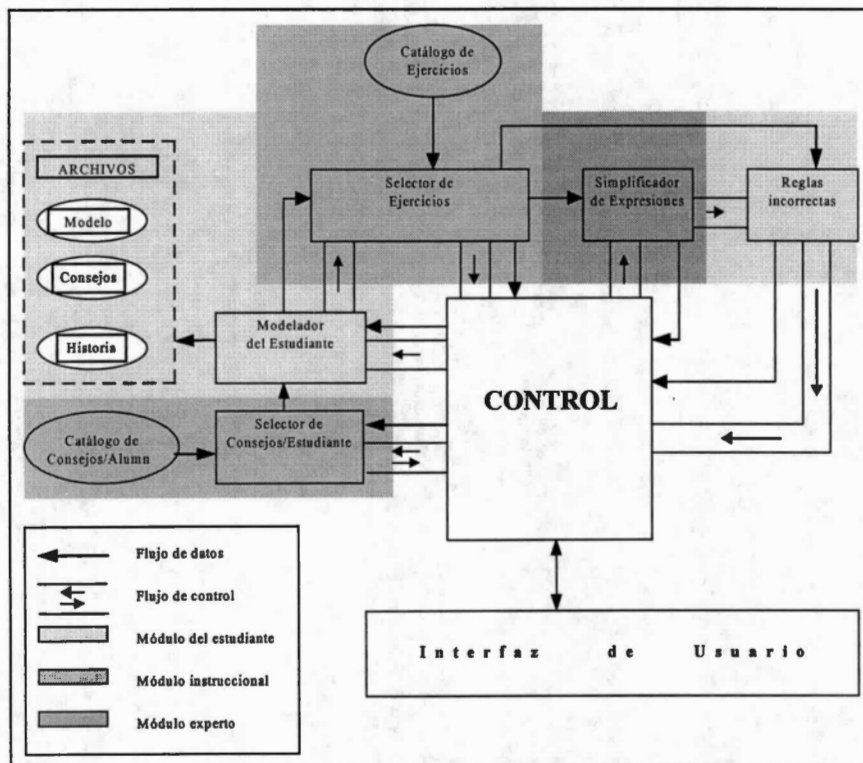


Figura 5-1. Estructura principal del sistema.

En la estructura mostrada arriba podemos observar la manera en que interaccionan los diferentes módulos que conforman el sistema para lograr la realización las operaciones

descritas en la sección *Actividades de aprendizaje* del capítulo cuatro y en el diagrama de flujo de la figura 4-2. En este diagrama vemos como el módulo de *Control* se encarga de distribuir el flujo de control entre los diferentes módulos del sistema. Primeramente el módulo de control cede el control al *Modelador del Estudiante* el cual lee el modelo del estudiante y lo transfiere al *Selector de Ejercicios* donde se crea el próximo ejercicio mediante el uso de una estructura base. El nuevo ejercicio pasa al módulo de *Control* para ser presentado al estudiante, y a los módulos *Simplificador de Expresiones* y *Reglas Incorrectas* para crear las opciones entre las que deberá elegir el estudiante. Los resultados obtenidos de estos dos últimos módulos son transferidos al módulo de *Control*. Mientras que el estudiante trabaja con un mismo ejercicio los módulos de *Control*, *Simplificador de Expresiones* y *Reglas Incorrectas* operan en forma cíclica transfiriéndose el control entre ellos. Cuando el ejercicio es resuelto por completo, el control se cede nuevamente al *Modelador del Estudiante* el cual evalúa su desempeño, actualiza el modelo de conocimientos del estudiante y registra sus errores.

## 5.1. Módulo Experto.

El módulo experto es el módulo que contiene el conocimiento sobre las leyes de los exponentes y lo aplica para la solución de problemas. Cumple una doble función dentro de los tutores inteligentes, por un lado, proporciona el conocimiento en el dominio particular, y por el otro, representa el estándar para la evaluación del estudiante. La elección del modelo para la representación del conocimiento es el aspecto más importante en el diseño del módulo experto. Debe ser el que mejor permita la solución de problemas e inferencias en el dominio específico del conocimiento.

Este módulo se basa en el modelo de *caja negra* con una base de *conocimiento procedural* (sección 3.1.1) y está diseñado usando una representación del conocimiento mediante reglas de producción. Cada regla en el módulo representa alguna ley de los exponentes o del álgebra en general. Para algunos casos fue necesario el diseño de más de una regla para obtener el resultado de aplicación de alguna ley, donde cada regla representa una etapa en la aplicación de una ley.

### 5.1.1. Simplificación de expresiones.

El orden en el cual se aplican las reglas o leyes de simplificación es esencial para obtener el resultado correcto. Cabe hacer notar que no es necesario realizar una búsqueda espacial de la solución dado las características de complejidad de las expresiones utilizadas y las leyes que nos interesa que se apliquen; por lo tanto es suficiente con aplicar la secuencia de reglas de simplificación siempre que se cumpla en la expresión.

El sistema realiza la simplificación de las expresiones aplicando las reglas de simplificación en el siguiente orden:

1. Simplificar al máximo el numerador
  - a. Convertir a *uno* variables, constantes y expresiones elevadas a la *cero*
  - b. Individualizar los exponentes de las variables y constantes eliminando paréntesis
  - c. Realizar la potenciación de constantes
  - d. Obtener una constante por término (multiplicación de constantes)
1. Simplificar al máximo el denominador
  - b. Se repiten los pasos del 1a al 1d sobre el denominador
1. Simplificar constantes del numerador y denominador al quebrado más simple
2. Eliminar variables, numerador contra denominador
3. Reacomodar variables en su nivel de acuerdo al signo del exponente
4. Reacomodar constantes y variables en cada término y nivel de acuerdo al orden: constante, x, y, z
5. Sumar términos semejantes.

El orden de los pasos de simplificación anterior indica el orden de precedencia de las reglas del módulo experto. Las reglas de simplificación que conforman el módulo experto se describen en la siguiente sección.

### 5.1.2. Especificación de reglas correctas de simplificación.

En esta sección daremos una descripción de las reglas diseñadas para la realización del algoritmo mencionado en la sección anterior. El símbolo  $\Rightarrow$  lo podemos interpretar como: “se asegura que...”. La figura 5-2 muestra una tabla que resume las características más importantes de cada regla.

#### **1.- regla: Potencia de un exponente *cero* (primera):**

convierte a *uno* las expresiones elevadas a exponentes *cero*. Ejemplo:

$$3x(4x^2y)^0z+5xy \Rightarrow 3x1z+5xy$$

*condiciones de aplicación:* busca un par de paréntesis que contengan un exponente *cero* en el paréntesis de cierre y, revisa que la expresión entre paréntesis no contenga el símbolo de paréntesis para asegurar que los paréntesis de entrada y salida se corresponden.

#### **2.- regla: Potencia de un exponente *cero* (segunda):**

convierte a *uno* las variables y constantes elevadas a la potencia *cero*. Ejemplo:

$$2^0xyx^0-3+5y^2 \Rightarrow 1xy1-3+5y^2$$



*condiciones de aplicación:* busca un símbolo elevado al exponente *ceros* y, revisa que la base no sea un paréntesis.

**3.- regla: Potencia de un exponente uno (primera):**

cuando encuentra una expresión elevada a la uno elimina el paréntesis y el exponente, Ejemplo:

$$(3x^2)^1 \rightsquigarrow 3x^2$$

*condiciones de aplicación:* busca un par de paréntesis que contengan un exponente *uno* en el paréntesis de cierre y, revisa que la expresión entre paréntesis no contenga el símbolo de paréntesis para asegurar que los paréntesis de entrada y salida se corresponden.

**4.- regla: Potencia de un exponente uno (segunda):**

cuando encuentra una variable o constante elevada a la unidad elimina el exponente. Ejemplo:

$$3x^1 \rightsquigarrow 3x$$

*condiciones de aplicación:* busca un símbolo elevado al exponente *uno* y, revisa que la base no sea un paréntesis.

**5.1.- regla: Segunda y tercera ley de los exponentes (primera):**

cuando encuentra un término elevado a una potencia, eleva la primer variable o constante del término a la potencia y lo separa del término. Sólo se ejecuta cuando la variable o constante no están elevadas a una potencia dentro del término. Ejemplo:

$$(3x^2)^2 \rightsquigarrow 3^2(x^2)^2$$

*condiciones de aplicación:* revisa que la expresión entre paréntesis no contenga los símbolos: +, -, ^, ni paréntesis.

**5.2.- regla: Segunda y tercera ley de los exponentes (segunda):**

trabaja en conjunto con la regla anterior para elevar el término a una potencia. Esta regla sólo se ejecuta cuando la variable o constante en término está elevada a una potencia. Ejemplo:

$$(3x^2)^2 \rightsquigarrow x^4(3)^2$$

*condiciones de aplicación:* revisa que la expresión entre paréntesis no contenga los símbolos: +, -, ^, ni paréntesis..

**5.3.- regla: Segunda y tercera ley de los exponentes (tercera):**

finalmente, esta tercera regla que concluye el trabajo de las últimas dos reglas descritas. Al final del trabajo de las dos reglas anteriores se obtienen los corchetes vacíos elevados a una potencia, esta regla se encarga de eliminarlo. Ejemplo:

$$()^2 \implies$$

*condiciones de aplicación:* ninguna.

**6.- regla: Potencia de una constante:**

ejecuta la operación aritmética de elevar constantes a su respectiva potencia. Ejemplo:

$$-3y4^3+5^2 \implies -3y64+25$$

*condiciones de aplicación:* ninguna.

**7.- regla: Simplificación de constantes en un término:**

multiplica pares de constantes en cada término. Se obtiene una constante por término después de varias ejecuciones. Ejemplo:

$$2x4y2x^2-3y4x^3+5y^2 \implies 8xy2x^2-12yx^3+5y^2$$

*condiciones de aplicación:* revisa que entre las dos constantes seleccionadas no estén los signos: +, -, ni paréntesis para asegurar que están dentro del mismo término, y que ninguna constante sea exponente.

**8.- regla: Primera ley de los exponentes:**

multiplica pares de variables del mismo tipo en cada término. Se obtiene una variable de cada tipo por término después de varias ejecuciones. Ejemplo:

$$16xyx^2-12xyx^3+5y^2 \implies 16x^3y-12yx^4+5y^2$$

*condiciones de aplicación:* busca una par de variables iguales y revisa que entre las dos variables seleccionadas no estén los signos: +, -, /, ni paréntesis, para asegurar que están dentro del mismo término. Determina el exponente al que está elevada cada variable para considerarlo en la multiplicación.

**9.1.- regla: Cuarta ley de los exponentes (primera):**

cuando encuentra dos variables iguales en el mismo término, una en el numerador y otra en el denominador, realiza la simplificación correspondiente restando sus respectivos exponentes. Ejemplo:

$$6x/zx^2-2x/zx^3+3z^4 \implies 6/zx-2/zx^2+3z^4$$

*condiciones de aplicación:* busca un par de variables iguales, una en el numerador y otra en el denominador y revisa que entre las dos variables seleccionadas no estén los signos: +, -, ni paréntesis, ni un signo extra de división, para asegurar que están dentro del mismo término. Determina el exponente al que está elevada cada variable para considerarlo en la simplificación. Si los exponentes son iguales elimina las variables del numerador y del denominador; si un exponente es mayor que otro, elimina la variable con exponente menor dejando una variable con exponente positivo en la posición en que se encontraba inicialmente con su exponente disminuido.

**9.2.- regla: Cuarta ley de los exponentes (segunda):**

cuando se detecta un símbolo divisor debajo del cual no existe denominador resultado de la aplicación de la regla anterior, se elimina. Sólo se aplica cuando le siguen otros símbolos. Ejemplo:

$$4zyx^2/-2xyx^5 \implies 4zyx^2-2xyx^5$$

*condiciones de aplicación:* esta situación se puede presentar después de aplicar la regla 12. revisa que al signo / le siga el signo + o el -.

**9.3.- regla: Cuarta ley de los exponentes (tercera):**

cuando se detecta un símbolo divisor debajo del cual no existe denominador resultado de la aplicación de la regla anterior, se elimina. Sólo se aplica cuando no le siguen otros símbolos. Ejemplo:

$$4zyx^2-2xyx^5/ \implies 4zyx^2-2xyx^5$$

*condiciones de aplicación:* esta situación se puede presentar después de aplicar la regla 12. No hace ninguna revisión extra.

**10.- regla: Acomodo de variables:**

reacomoda un par de variables dentro de cada término siguiendo el orden: x, y, z. El reacomodo lo realiza mediante aplicaciones sucesivas de la regla. Ejemplo:

$$7xzy^2-3zyx^3+5yx^2 \implies 7xy^3z-3x^3yz+5x^2y$$

*condiciones de aplicación:* busca un par de variables donde la primera variable sea una “y” y la segunda una “x” o donde la primera sea una “z” y la segunda una “y”. Si las encuentra las invierte de posición junto con sus respectivos exponentes. Para asegurar que pertenezcan al mismo término se revisa que entre ellas no estén los signos: +, -, /, ni paréntesis.

**11.- regla: Acomodo de constantes:**

tiene como propósito el colocar la constante al inicio de cada término. Ejemplo:

$$xy8x^2-4xyx^3 \rightsquigarrow 8xyx^2-4xyx^3$$

*condiciones de aplicación:* busca una constante que se encuentre después que alguna variable en la expresión y las invierte de posición junto con el exponente y, revisa que no estén separadas por los signos: +, -, /, ni paréntesis.

**12.1.- regla: Suma algebraica (primera):**

es la primera de una serie de reglas destinadas a realizar la operación de suma algebraica. Esta regla en particular tiene como propósito el descomponer la expresión en términos. Se aplica al momento de iniciar la separación de términos, por lo que sólo se aplica una vez por expresión. Ejemplo:

$$16x^3y+12yx^4+5y^2+2yx \rightsquigarrow 16x^3y, +12yx^4+5y^2+2yx$$

*condiciones de aplicación:* busca que la expresión no inicie con los signos: +, ni -. Luego toma una subexpresión desde el primer elemento hasta donde encuentre el primer signo + o -; revisa que esa subexpresión no contenga paréntesis. Como resultado se obtienen dos hechos la expresión original sin el primer término y un término independiente. Si la expresión tenía inicialmente dos términos, el resultado son los dos términos independientes, y la expresión se elimina. Cada término generado tiene un identificador para el caso en que existan términos exactamente iguales.

**12.2.- regla: Suma algebraica (segunda):**

esta regla tiene la función de separar en términos al igual que la regla anterior. Se aplica cuando se ha iniciado ya la separación en términos por la regla anterior. Se encarga de separar los términos restantes, el resultado se obtiene por aplicación sucesiva de la regla. Ejemplo:

$$+12yx^4+5y^2+2yx \rightsquigarrow +12yx^4, 5y^2, +2yx$$

*condiciones de aplicación:* son las mismas que el caso anterior, excepto que revisa que la expresión inicial tenga en su primer elemento el signo :+ o -, lo cual indica que ya se le separo su primer término de la expresión inicial.

**12.3.- regla: Suma algebraica (tercera):**

tiene como función comparar los términos ya separados por las dos reglas anteriores, determinar si cumplen las condiciones para sumarse, y realizar la suma. Ejemplo:

$$3x^2, 2x^2, 5y^2 \rightsquigarrow 5x^2, 5y^2$$

*condiciones de aplicación:* busca dos términos con identificador diferente y que su base sea igual y realiza la suma o resta de acuerdo al valor de la constante y a su signo.

**12.4.- regla: Suma algebraica (cuarta):**

su propósito es el de unir los términos separados en una solo expresión, terminando así la operación de la suma algebraica. Ejemplo:

$$16xyx^2, -12xyx^2, 5y^2 \rightsquigarrow 16x^3y-12yx^4+5y^2$$

*condiciones de aplicación:* ninguna. Elimina los términos a medida que los une a la expresión.

NO.	DESCRIPCIÓN DEL PRINCIPIO ALGEBRAICO	EJEMPLO	EXPLICACIÓN
1.	Potencia de un exponente <i>cero</i> de una expresión	$(ABC)^0 = 1$	Convierte a <i>uno</i> las expresiones elevadas al exponente <i>cero</i>
2.	Potencia de un exponente <i>cero</i> de una variable o constante	$A^0 = 1$	Convierte a <i>uno</i> la expresión elevada al exponente <i>cero</i>
3.	Potencia de un exponente <i>uno</i> de una expresión	$(ABC)^1 = ABC$	Cuando se encuentra una expresión elevada al exponente <i>uno</i> elimina el paréntesis y el exponente
4.	Potencia de un exponente <i>uno</i> de una variable o constante	$A^1 = A$	Cuando se encuentra una variable o una constante elevada al exponente <i>uno</i> elimina el exponente
5.	Segunda y tercera ley de los exponentes	$(AB)^n = A^n B^n$ $(A^m)^n = A^{m \cdot n}$	Eleva la expresión al exponente <i>n</i> multiplicando los exponentes de cada variable o constante del término por el exponente de la expresión
6.	Potencia de una constante	$2^3 = 8$	Ejecuta la operación aritmética de elevar constantes a su respectiva potencia
7.	Simplificación de constantes en un término	$8x3y = 24xy$	Multiplica constantes en cada término
8.	Primera ley de los exponentes	$A^n A^m = A^{n+m}$	Multiplica variables iguales en cada término
9.	Cuarta ley de los exponentes	$A^n / A^m = A^{n-m}$	Cuando encuentra dos variables iguales en el mismo término, una en el numerador y otra el denominador, realiza la simplificación correspondiente restando sus respectivos exponentes
10.	Acomodo de variables	$yxz = xyz$	Reacomoda variables dentro de cada término siguiendo el orden: x, y, z
11.	Acomodo de constantes	$xy8z = 8xyz$	Coloca la constante al inicio de cada término
12.	Suma algebraica	$2A+3A = 5A$	Suma términos semejantes dentro de la expresión

**Figura 5-2. Reglas de simplificación de expresiones.**

## 5.2. Módulo del estudiante.

Este módulo determina qué sabe y qué no sabe el estudiante. Para llegar a conocer esto, el módulo puede apoyarse en: (1) desempeño del estudiante al intentar resolver problemas, (2) los datos históricos que permiten medir su nivel de habilidad, y (3) el nivel de dificultad de los ejercicios.

El sistema que se describe en este trabajo contiene un *módulo de diagnóstico del estudiante* que se apoya en un conjunto de reglas de producción similares a las reglas de simplificación, pero que arrojan resultados de simplificación incorrectos. Estas reglas de producción son variantes lógicas de las reglas correctas y representan errores que el estudiante puede cometer después de un razonamiento lógico acerca del ejercicio que debe resolver y las herramientas de simplificación con que cuenta, estas reglas se describen en un sección más adelante dentro de este capítulo. Los aciertos y desaciertos del estudiante produce la actualización de un registro que representan el *modelo del estudiante*, el cual se analiza con más detalle en la siguiente sección. El ancho de banda empleado en esta técnica es el de *estados intermedios* (sección 3.1.2) ya que se dispone de algunos estados mentales significativos del estudiante durante la solución de ejercicios.

### 5.2.1. Modelo del estudiante.

El sistema maneja un conjunto de reglas de simplificación correctas y otro conjunto de reglas de simplificación incorrectas. Las reglas correctas representan al modelo del experto y se han descrito con anterioridad (sección 5.1.1). En contraste, las reglas incorrectas han sido incluidas para poder modelar el conocimiento erróneo del estudiante (sección 5.2.2).

Para realizar la modelación del conocimiento del estudiante se pensó en el uso de la lógica difusa y la asignación de un peso a cada concepto o conocimiento, que indique el grado de dominio que tiene el estudiante en el manejo de ese concepto. Gisolfi, Dattolo y Balzano [Gisolfi, Dattolo y Balzano, 1992], mencionan que el nivel de conocimiento o aprendizaje es un concepto inherentemente vago e impreciso, por lo que si se apoyara la modelación del estudiante sobre la teoría de la lógica difusa resultaría ampliamente beneficiada, dado que el propósito de la lógica difusa es el manejo de este tipo de situaciones. Esta técnica es una variante de la técnica del *catálogo de errores* (sección 3.1.2), pero con asignación de pesos a cada elemento de la base de conocimientos.

Para acoplar la lógica difusa a la modelación del estudiante se considera un modelo del estudiante como un conjunto de reglas correctas y un peso difuso asociado a cada regla. El aspecto más crítico en el uso de la lógica difusa para la modelación del estudiante es la determinación de la evolución de los pesos durante la sesión de ejercicios. Sería inapropiado en el caso de los ejercicios de álgebra, hablar de ejercicios bien o mal resueltos,

cada ejercicio involucra más de una ley del álgebra y debe hacerse, más bien, una valoración para cada una de ellas.

Como describimos en la sección *especificación funcional* (sección 4.1.1), la sesión de ejercicios el sistema plantea una expresión algebraica al estudiante y le propone entre 3 y 5 posibles rutas de simplificación mediante la ejecución de reglas correctas e incorrectas, el estudiante debe seleccionar la ruta que considera más apropiada, es decir, aquella que sea resultado de la aplicación de una regla correcta y que conduzca más rápido a la simplificación de la expresión. Cada ejercicio se relaciona con un conjunto de leyes de simplificación y los pesos difusos de estas leyes se modifican de acuerdo a las selecciones del estudiante.

Cada regla correcta se relaciona con un conjunto de reglas incorrectas que representan los errores comunes al aplicar esa regla correcta, igualmente, la relación entre reglas correctas e incorrectas puede verse a la inversa, es decir, para cada regla incorrecta existe un conjunto de reglas correctas relacionadas con ella. La figura 5-3 muestra estas relaciones entre las reglas correctas e incorrectas. La manera en que evolucionan los pesos difusos de las reglas se explica enseguida:

las reglas correctas:

1. aumentan su peso cada vez que es seleccionada por el estudiante la expresión resultante de su aplicación de entre las opciones expuestas por el sistema,
2. bajan su peso cuando se selecciona otra expresión diferente a la correcta de las expresiones expuestas por el sistema, y
3. también bajan su peso cuando se selecciona una regla incorrecta relacionada con ella.

Las reglas incorrectas:

1. aumentan su peso cuando son seleccionadas, y
2. bajan su peso cuando se selecciona una regla correcta relacionada con ella.

Num. de regla correcta/incorrecta	Regla correcta relacionada	Regla incorrecta relacionada
1	1 a 4	1 a 3
2	1 a 4	1 a 3
3	1 a 4	1 a 3
4	5 a 7	1 a 3
5	5 a 7	4, 5
6	7	4, 5
7	7	4 a 8
8	7	9, 10, 13
9	8	10 a 12
10	8, 9	--
11	9	--
12	9	13
13	8, 12	

Figura 5-3. Relaciones entre reglas correctas e incorrectas.

El peso difuso de calcula introduciendo una función que varía en el intervalo (0,1) y cuyos argumentos son el peso difuso actual y un valor  $k$  de la respuesta. Esta función hace que el peso difuso de las reglas se incremente o decremente siguiendo el mecanismo expuesto arriba.

El peso difuso está definido en la siguiente función recursiva:

$$w':(w, k) \rightarrow \begin{cases} w^{1/2^k} & 0 < k \leq i \\ w^{2^{|k|}} & j \leq k, w \in (0, t) \\ w^{|k|} & j \leq k, w \in [t, 1) \end{cases}$$

Donde:

$n$ , es el número de posibles respuestas asociadas con cada paso de la simplificación,  
 $i$ , subconjunto de las  $n$  respuestas consideradas correctas,  
 $j$ , es igual a  $i - n$ , por lo que tendrá un valor negativo,  
cada repuesta tiene asociada una constante entera  $k$  entre el rango (0,  $i$ ) para las respuestas correctas, y en el rango ( $j$ , 0) para las respuestas incorrectas. El valor  $k$  tiene como propósito darle un peso a cada respuesta de acuerdo a la importancia para los objetivos instruccionales del sistema, un valor de  $k$  más grande acelera la velocidad de cambio de  $w$ .

Cuando la respuesta es correcta se usa la primer versión de la función de  $w$ , cuando es incorrecta y el valor actual de  $w$  es menor a  $t$ , se usa la segunda versión y, cuando la respuesta es incorrecta y el valor actual de  $w$  es mayor que  $t$  se usa la tercer versión.



El parámetro  $t$  tiene un valor en el rango  $(0, 1)$  y su valor es tal que el intervalo  $[t, 1)$  indica el rango de valores de  $w$  para el cual se asume que el estudiante domina un concepto específico. De igual forma el parámetro  $s$  cae dentro del rango  $(0, 1)$  y su valor es tal que el intervalo  $(0, s)$  indica el rango de valores de  $w$  para el cual se asume que el estudiante no domina un concepto específico. En consecuencia se puede hablar de la existencia de tres niveles de aprendizaje del estudiante los cuales se obtienen de la partición en tres niveles del intervalo  $(0, 1)$ :

- Si  $w$  pertenece al rango  $(0, s)$ , se concluye: *sin conocimiento*
- Si  $w$  pertenece al rango  $[s, t)$ , se concluye: *no se puede asegurar*
- Si  $w$  pertenece al rango  $[t, 1)$ , se concluye: *conocimiento*

El sistema lleva un registro del conocimiento del estudiante relacionando cada regla mediante una etiqueta correspondiente al nivel de aprendizaje.

De acuerdo a las características del sistema descrito en este trabajo, se seleccionaron los siguientes valores para los parámetros de la expresión recursiva para el peso difuso:

$k = 2$ , constante asociada a las respuestas correctas,

$k = -2$ , constante asociada a las respuestas incorrectas,

$s = 0.1$ , límite de  $w$  debajo del cual se considera que el alumno no domina una ley,

$t = 0.9$ , límite de  $w$  encima del cual se considera que el alumno si domina una ley.

Se seleccionaron estos valores porque haciendo una tabulación para los valores de  $w$  usando estos parámetros se encontró que en promedio se requieren tres aciertos continuos para que una ley pase de ser considerada “no dominada” a ser considerada “dominada” por el estudiante. Además, Gisolfi menciona que a partir del rango  $[0.7, 0.9]$  los valores de  $w$  no varían mucho, por lo que recomienda que el valor de  $t$  se encuentre dentro de ese rango. Por otro lado, para el parámetro  $s$  propone valores menores a 3.5.

### **Proceso de modelación del estudiante.**

El proceso utilizado en este sistema resulta ser más simple que el propuesto por Gisolfi, Dattolo y Balzano [Gisolfi, Dattolo y Balzano, 1992], ya que para nuestro caso no es necesario buscar la fuente de error dado que va implícita en la respuesta del estudiante y además nunca habrá más de una fuente de error posible, en consecuencia el proceso queda descrito de la siguiente forma:

Representamos por  $p$  la regla correspondiente que dio como resultado la opción seleccionada por el estudiante.

- Se computa el peso difuso  $w'$  para  $p$ .
- Si el valor correspondiente  $w \in [t, 1)$ , entonces  $p$  es almacenado con la etiqueta *conocimiento*.
  - Si el valor correspondiente  $w \in [s, t)$ , entonces  $p$  es almacenado con la etiqueta *no se puede asegurar*.
  - Si el valor correspondiente  $w \in (0, s)$ , entonces  $p$  es almacenado con la etiqueta *sin conocimiento*.

### 5.2.2. Especificación de reglas incorrectas.

Como se ha mencionado en secciones anteriores, el sistema aplica reglas incorrectas que le permiten valorar el conocimiento del estudiante. En esta sección se da una descripción de cada regla incorrecta definida. Empezaremos ofreciendo una explicación del fundamento teórico de la elección de las reglas incorrectas.

El sistema ejecuta reglas erróneas además de las reglas correctas porque requiere un punto de referencia para juzgar el desempeño del estudiante. El estudiante debe elegir entre opciones que surgen de reglas correctas y opciones que surgen de reglas incorrectas, de esa manera, si el estudiante optó por la opción surgida de una regla incorrecta, puede evaluarse el tipo de error cometido y el conocimiento faltante o erróneo que posee el estudiante.

Cuando un estudiante comete errores en el manejo del álgebra, es debido a que no ha utilizado correctamente un conocimiento previo, o bien, no ha sabido aplicar correctamente una técnica de extrapolación. Matz dice que los errores en álgebra son el resultado de un intento sin éxito de usar las reglas conocidas a un problema nuevo. La solución de problemas tiene dos componentes: (1) el conocimiento que posee el estudiante y, (2) el manejo de la técnicas de extrapolación [Matz, 1982].

Los errores cometidos tienen un fundamento razonable debido a que las técnicas de extrapolación usadas se aplican correctamente a ejemplos que son una variación en la forma del ejercicio prototipo del cual se extrajo. Matz menciona tres fuentes posibles de error: (1) por la elección incorrecta de la *técnica de extrapolación*, (2) por una *base de conocimientos* pobre y, (3) errores de *procedimiento*.

La técnica de *extrapolación de linealidad* es aquella que permite trabajar con un objeto compuesto dividiéndolo en sus partes y tratando cada parte como un objeto independiente. Un operador se aplica linealmente cuando se aplica a cada parte de una expresión y se unen los resultados parciales. La descomposición lineal es correcta para algunos casos y para otros no. Los ejemplos correctos e incorrectos pueden generalizarse usando los siguientes esquemas:

para un operador binario ( $\Delta$ ) y uno unitario ( $\Omega$ ):

ESQUEMA 1:  $\Omega (x \Delta y) \Rightarrow \Omega x \Delta \Omega y$

para dos operadores binarios:

ESQUEMA 2:  $(x \Delta y) \Omega z \Rightarrow (x \Delta z) \Omega (y \Delta z)$

ESQUEMA 3:  $x \Delta (y \Omega z) \Rightarrow (x \Delta y) \Omega (x \Delta z)$

Por su lado la técnica de **extrapolación por generalización** consiste en el ajuste de un regla conocida para que pueda aplicarse a una nueva expresión. Los errores clásicos en esta categoría surgen porque el estudiante llega a considerar sin importancia los números que aparecen en una regla algebraica y cuando los números de la regla son críticos. Ejemplos de este tipo son las leyes de identidad para adición y multiplicación:

$$A * 1 = A \quad \text{y} \quad A + 0 = A$$

Para un operador binario ( $\Delta$ ), el estudiante puede generalizar:

$$A \Delta (\text{número especial}) = A$$

y en base a esto pueden aparecer los siguientes errores:

$$A * 0 \Rightarrow A \quad \text{y} \quad A * 1 = A$$

donde el símbolo “ $\Rightarrow$ ” representa una igualdad errónea.

Igualmente para el caso de la adición y multiplicación del inverso:

$$A + (-A) = 0$$

generalizado:

$$A \Delta (\text{inverso de } A) = 0$$

regla errónea surgida:

$$(1/A) \Rightarrow 0$$

Los cambios conceptuales que implica resolver problemas de álgebra a partir del conocimiento de la aritmética, trae consigo otros cuantos motivos para cometer errores al resolver problemas de álgebra. La concatenación es uno de esos cambios conceptuales en la transición de la aritmética al álgebra. Por ejemplo, en la aritmética puede usarse la concatenación como adición implícita:

71/2

que representan la suma de “50 +7”, y “7+1/2”, en cambio, la concatenación en álgebra representa la multiplicación:

xy

5x

por lo tanto habrá casos en los cuales el estudiante sumará números que realidad debieran multiplicarse.

De la discusión anterior acerca del origen de muchos de los tipos de errores se diseñaron las reglas incorrectas para el diagnóstico del estudiante, estas reglas se describen abajo en la figura 5-3.

NO.	REGLA INCORRECTA:	REGLAS CORRECTAS RELACIONADAS:	FUNDAMENTO EN EL ESTUDIO DE MATZ:
1	$A^0 \Rightarrow 0$	reglas 1 y 2	Extrapolación por generalización
2	$A^0 \Rightarrow A$	reglas 1 y 2	Extrapolación por generalización
3	$A^1 \Rightarrow 1$	reglas 3 y 4	Extrapolación por generalización
4	$(A+B)^C \Rightarrow A^C+B^C$	reglas 5	Linearidad
5	$(A^B)^C \Rightarrow A^{B+C}$	reglas 5	Linearidad
6	$A*0 \Rightarrow A$	regla 7	Extrapolación por generalización
7	$A*1 \Rightarrow 1$	regla 7	Extrapolación por generalización
8	$3x4y \Rightarrow 7xy$	regla 7	Cambio conceptual
9	$2xyx \Rightarrow 3xy$	reglas 8	Cambio conceptual
10	$A^B A^C \Rightarrow A^{BC}$	reglas 8 y 9	Linearidad
11	$A^B/A^C \Rightarrow A^{B/C}$	regla 9	Linearidad
12	$A/A \Rightarrow 0$	regla 9	Extrapolación por generalización
13	$A^B+A^C \Rightarrow A^{B+C}$	reglas 8 y 12	Linearidad

Figura 5-4. Reglas de simplificación incorrectas.

Además de las reglas incorrectas mencionadas, otra fuente de errores es la aplicación de reglas correctas en un orden incorrecto y que por ese motivo no conducen hacia una expresión más simple.

Dado que las reglas incorrectas se aplican bajo las mismas condiciones que las reglas correctas relacionadas, y la única diferencia será el resultado que arroja, no es necesario detallarlas como en el caso de las reglas correctas.

Finalmente cabe mencionar que las reglas extraídas del estudio teórico de Matz [Matz, 1982] coincide en gran medida con los errores más frecuentes encontrados en la práctica por Margarita Toro [Toro, 1996].

### **5.3. Módulo instruccional.**

El *módulo instruccional* es el encargado de diseñar la estrategia instruccional de acuerdo a las valoraciones sobre el estudiante. Este módulo identifica las diferencias de conocimiento entre el estudiante y el experto para luego seleccionar una estrategia instruccional adecuada. Además, puede estar provisto con capacidad para responder a las preguntas del estudiante y ayudarlo cuando enfrente dificultades [Burns y Capps, 1988].

En el caso del sistema propuesto en este trabajo, el módulo instruccional se apoyará en el *modelo del estudiante*, descrito en la sección con el mismo nombre en este capítulo (5.2.1), para seleccionar el ejercicio más apropiado que se le presentará al estudiante. Otra de las funciones que cumplirá es la de aconsejar al estudiante durante la sesión de ejercicios basándose en las respuestas que el usuario genere. Toda la información del desempeño del estudiante se mantendrá actualizada en un archivo específico para cada estudiante, que podrá ser consultado por el alumno o su maestro.

#### **5.3.1. Selección de ejercicios y niveles de dificultad.**

Para la selección de ejercicios se hace uso del modelo del estudiante. Los ejercicios se toman de un catálogo de estructuras de ejercicios, donde las estructuras están agrupadas por las principales leyes del álgebra que involucra su solución. Cada grupo de estructuras está etiquetada por una serie de índices que representan las leyes que es necesario aplicar para solucionarlo. El mecanismo de selección consiste en generar los índices tomando como entrada el modelo del estudiante, realizar una búsqueda en el catálogo de ejercicios del índice, seleccionar una estructura aleatoriamente del grupo correspondiente y, la estructura seleccionada se convierte en una expresión algebraica por la generación de las constantes (coeficientes y exponentes) de manera aleatoria.

De acuerdo a lo descrito en el párrafo anterior el próximo ejercicio seleccionado será aquel que para su solución se ejecute sólo una ley que se supone “no conocida” por el estudiante. La razón de lo anterior es que si el ejercicio sólo involucra el conocimiento que ya posee el estudiante su solución no aportaría nada nuevo, por otro lado, si se selecciona un ejercicio que presente más de una ley considerada “no conocida”, será más difícil

determinar las fuentes reales de error. La figura 5-4 muestra el procedimiento de selección de próximo ejercicio que se propondrá al estudiante. De esta manera, el nivel de dificultad del ejercicio seleccionado estará dado por las reglas conocidas por el estudiante. Por lo descrito anteriormente, la selección de los ejercicios propuestos al estudiante busca cumplir de la mejor manera posible con la mayoría de los principios instruccionales descritos por Halff [Halff, 1988] (sección 3.1.3), es decir, *manejabilidad, individualización, divergencia y transparencia estructural*.

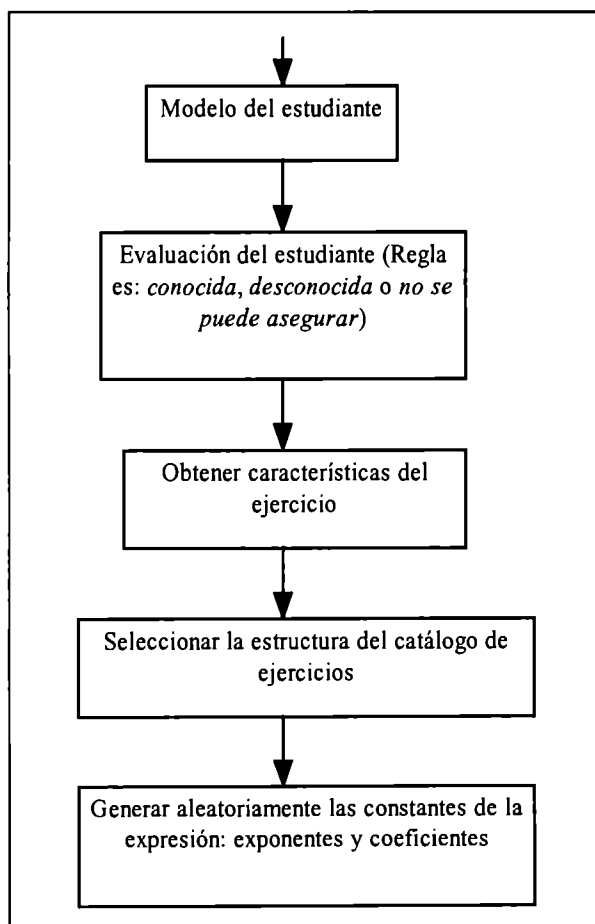


Figura 5-5. Procedimiento para la selección de ejercicios.

### 5.3.2. Consejos para el estudiante.

Como hemos explicado en secciones anteriores el sistema de tutorío está provisto con la capacidad para ofrecer consejos a los estudiantes después de cada ejercicio. Los consejos que ofrece el sistema son acordes al tipo de error detectado en el desempeño del estudiante.

Los consejos se toman de un *catálogo de consejos* y se seleccionan de acuerdo a la trayectoria de simplificación seguida por el estudiante durante la solución del último ejercicio. Estos consejos fueron diseñados tomando como base los resultados obtenidos en una encuesta a maestros de matemáticas de la preparatoria Eugenio Garza Sada del ITESM (ver anexo), cuyos resultados se encuentran en las conclusiones de este trabajo, y las bases de los procesos cognitivos en el diseño de material instruccional [Toro, 1996]. A continuación se presenta un ejemplo de los consejos para algunos tipos de error.

Regla incorrecta 1	$A^0 \Rightarrow 0$ Incorrecto $A^0 = 1$ Correcto  Explicación:  El exponente <i>ceros</i> proviene de la división de potencias de la misma base:  $A^2 / A^2 = 1 = A^{(2-2)} = A^0 = 1$  Toda cantidad elevada a la <i>ceros</i> equivale a 1, ya que toda cantidad dividida por sí misma es igual a 1.
Regla incorrecta 3	$A^1 \Rightarrow 1$ Incorrecto $A^1 = A$ Correcto  Explicación:  El exponente indica las veces que la base es tomada como factor, así:  $A^2 = A * A$ $A^3 = A * A * A$ $A^1 = A$  Toda cantidad elevada al exponente 1 es igual a la misma cantidad.
Regla incorrecta 5	$(A^B)^C \Rightarrow A^{(B+C)}$ Incorrecta $(A^B)^C = A^{(B*C)}$ Correcto  Explicación:  El exponente indica las veces que la base es tomada como factor, así:  $(A^2)^3 = (A^2)*(A^2)*(A^2) = (A*A)*(A*A)*(A*A) = A*A*A*A*A*A = A^6$  Segunda Ley: La potencia de una potencia es igual a la base elevada a multiplicación de los exponentes.

Figura 5-6. Tabla que muestra algunos de los consejos presentados por el sistema.

Puede diseñarse más de un consejo para cada tipo de error con diferentes ejemplos o estructura para permitirle al estudiante adquirir el conocimiento más fácilmente.

# Capítulo 6. Conclusiones y trabajos futuros.

## 6.1. Conclusiones.

### 1. Logros alcanzados.

El presente trabajo alcanzó logros significativos pues en gran medida se partió sólo de estudios teóricos para el diseño de la estructura general del sistema y de cada uno de los módulos que lo componen, no era tarea fácil contando con tan poco tiempo. Sin embargo, el diseño resultante es capaz de ofrecer un diagnóstico del conocimiento del estudiante, y presentarse como un medio para la práctica de ejercicios en un área como el álgebra donde esta labor es de hecho complicada. Entre los logros más importantes se encuentran:

1. La integración de estudios teóricos en el área de los sistemas de tutorio inteligente para alcanzar un diseño con las siguientes ventajas:
  - a. es posible diagnosticar el conocimiento del estudiante en un dominio donde esta actividad presenta dificultades dentro de la enseñanza tradicional, ya que se requiere dedicar tiempo para analizar el desempeño particular de cada estudiante dentro del aula,
  - b. representa una herramienta para la práctica de ejercicios de simplificación de expresiones. Los estudiantes cuentan con un medio novedoso, que no los deja solos durante la práctica de ejercicios, que les marca sus errores y les ofrece consejos para corregirlos,
  - c. por la estructura, las actividades de aprendizaje y la delimitación de la base de conocimiento para cubrir sólo los aspectos relevantes del dominio de interés, se logró un diseño sencillo para su implementación.



d. al proponerle opciones al estudiante se reduce la complejidad del sistema, no existen errores de edición de parte del estudiante y además la selección de una opción lleva implícito el error cometido, lo cual facilita el diagnóstico del estudiante.

2.- Como resultado del análisis del artículo de Matz [Matz, 1982], se crearon doce reglas de simplificación que producen resultados incorrectos. Cada una de estas reglas está relacionada con un error potencial que el estudiante puede cometer. El uso de estas reglas facilita la identificación del conocimiento faltante o erróneo que posee el estudiante. Además, están sustentadas sobre bases sólidas relativas a estudios sobre las principales fuentes de error en el álgebra.

3.- En base a los cuestionarios aplicados a maestros de matemáticas y apoyándose en estudios sobre los procesos cognitivos se diseñaron los consejos que el sistema ofrece a los estudiantes al cometer un error. En el capítulo cinco se analiza lo referente a los consejos para el estudiante y en el anexo aparece la relación completa de consejos diseñados.

4.- Se obtuvo un primer bosquejo de interfaz de usuario como resultado de la metodología descrita en el capítulo cuatro basada en el estudio de los usuarios potenciales.

5.- El sistema diseñado para este trabajo de tesis ha sido implementado utilizando un Shell de sistemas expertos desarrollado por la NASA, llamado Clips [Riley, 1997], y cuyo listado se encuentra en el anexo de esta tesis.

## **2. Resultados de la encuesta a usuarios potenciales.**

De la aplicación de una encuesta a los usuarios potenciales del sistema - alumnos y maestros de matemáticas de la preparatoria Eugenio Garza Sada del ITESM - se obtuvieron los siguientes resultados:

Los alumnos:

- Prefieren practicar sus ejercicios de álgebra en el salón de clases en vez de practicar en su casa o en la biblioteca,
- Los *alumnos regulares* prefieren practicar sus ejercicios solos o con sus compañeros de clases, y los *irregulares* prefieren practicarlos en presencia de su maestro de matemáticas,
- Cuando tienen alguna duda prefieren preguntarle al maestro antes que consultar un libro o preguntarle a un compañero de clase.

Los maestros:

- Al enseñar consideran que es más importante conocer el conocimiento erróneo del estudiante que su conocimiento correcto, pero no descartan a ninguno de los dos,
- Para detectar las causas de los errores de los estudiantes en la solución de ejercicios de álgebra proponen ejercicios al estudiante que involucran la aplicación de reglas que son frecuentes causas de error, de acuerdo a sus experiencias. Aprovechan para tal efecto las tareas, los exámenes y los ejercicios en clase,
- Para señalar un error al estudiante, resaltan el lugar del error, muestran la forma correcta de resolver el ejercicio y ofrecen ejemplos.

El diseño de las encuestas puede consultarse en el anexo de tesis.

## **6.2. Trabajos futuros.**

### **1. Mejorar la base de conocimiento.**

Uno de los módulos de mayor relevancia de un sistema de tutorio inteligente es sin duda el módulo experto, ya que es el punto de referencia para reconocer el poder instruccional del tutor inteligente, pero también para nuestro caso, de igual importancia es el módulo de reglas incorrectas que nos permiten diagnosticar al estudiante, ambos descritos en el capítulo cinco. Es importante proponer mejoras en estos módulos que permitan fortalecer el sistema de tutorio.

Una propuesta puede ser el rediseño de las reglas de los módulos experto y de reglas incorrectas para que el nivel de granularidad del conocimiento permita una detección más precisa de las fuentes posibles de error. Para lograr lo anterior se puede realizar previamente un análisis conceptual para detectar todas las posibles fallas de los estudiantes [Rico, 1993]. Esto puede hacerse en forma práctica, tomando, para cada uno de los conceptos que se espera que el estudiante aprenda, todos los conceptos y procedimientos que se le relacionan o que son prerrequisitos del concepto principal.

Otra mejora importante a la base de conocimiento lo constituye la creación de reglas de simplificación más flexibles al crecimiento continuo de la base de conocimiento. Esto es, la reformulación de la representación del conocimiento, por una representación que permita este crecimiento. Las reglas actuales del sistema pueden simplificar expresiones de complejidad limitada (ver capítulo cuatro), por lo que no es tan sencillo usarlo como base para el resto de las áreas del álgebra o para un nivel superior al de bachillerato.

## **2. Mejoras en la implementación e interfaz.**

El prototipo actual fue creado con una herramienta para el desarrollo de sistemas expertos, Clips [Riley, 1997]. Este sistema no es ejecutable independientemente, necesita el uso del sistema Clips como intérprete, lo cual lo hace poco práctico. Sin embargo, el sistema puede ejecutarse bajo diferentes plataformas computacionales, siempre que se cuente con la versión de Clips que se ejecute en esa plataforma.

Una excelente mejora puede ser el llevar el sistema al WWW para que pueda ser utilizado desde cualquier lugar del mundo. Existen dos mecanismos que pueden facilitar esta meta: (1) usar el lenguaje Java [Sun Microsystems Inc., 1997] apoyado por el sistema JESS [Freidman-Hill, 1997], el sistema podría ser ejecutado bajo cualquier plataforma; (2) usar el intérprete WebCLIPS [Giordano, 1997] desde un servidor de Web dentro de una PC.

Por otro lado, el prototipo actual tiene una interfaz complementada en texto y en blanco y negro. Esta forma de presentación no es la más adecuada para un sistema educativo. Un estudiante joven necesita ver colores y movimiento, es decir, una interfaz gráfica y mecanismos de interacción basados en la manipulación directa. Este trabajo contiene un primer estudio para la creación de una interfaz de usuario acorde a la actividad educativa, la cual está contenida en el capítulo cuatro, antes de ser implementada sobre el prototipo propuesto se requiere hacer pruebas y rediseñarla apropiadamente.

El lenguaje Java y el WebCLIPS proporcionan funciones para la creación de interfaces de usuario elegantes y funcionales, pero se puede crear una mejor presentación del sistema usando el Visual C++ o el Visual Basic apoyado por la Biblioteca de Enlace Dinámico (DLL) para Clips [Anthena Information Service, 1997]. Todas estas propuestas no requieren reescribir el código que se presenta en el anexo de este escrito.

## **3. Realizar una evaluación del sistema propuesto.**

El sistema requiere de una evaluación en sus características instruccionales y de apoyo al maestro en los diagnósticos del estudiante. Uno de los trabajos de mayor urgencia es sin duda la evaluación del sistema bajo condiciones de uso y con usuarios reales. Littman y Soloway [Littman y Soloway, 1998] presentan algunos estudios sobre la evaluación de sistemas de tutorio inteligente que pueden ser útiles para este propósito.

## **4. Propuesta de investigación.**

Como se ha mencionado en este escrito, para desarrollar sistemas de tutorio inteligentes con todas las cualidades inteligentes deseables, ya mencionadas en el capítulo

uno y más extensamente en el capítulo tres, se requiere de un gran esfuerzo. Sin embargo, ese esfuerzo se ve recompensado por las ventajas educativas obtenidas a partir del uso de este tipo de sistemas:

- instrucción más individualizada,
- flexibilidad de horarios y tiempos de instrucción,
- facilidad de diagnóstico y seguimiento del progreso del estudiante,
- posibilidad de usarlo como medio de instrucción a larga distancia, entre otras.

Por lo antes mencionado, no sería tiempo perdido trabajar en un programa de desarrollo de sistemas de tutorio inteligentes a largo plazo, buscando como primer objetivo diseñar una estructura del tutor inteligente que en la medida de lo posible permita la flexibilidad de uso en diferentes áreas, el mantenimiento y el reuso de código, en lugar de trabajar con módulos expertos independientes que en muchos casos pueden ser difíciles de adaptar al propósito de la instrucción. Un área propicia para iniciar son las matemáticas, ya que por tradición los alumnos enfrentan obstáculos, a veces difíciles de detectar, que no les permiten adquirir el conocimiento.

# Anexos.

## Anexo 1. Listado del módulo Simplificador del sistema.

```
(defmodule SIMPLIFICADOR
  (import MAIN deftemplate initial-fact)
  (import CONTROL2 deftemplate expres_nueva)
)

(defrule SIMPLIFICADOR::recibe_expression_nueva
  ?antes <- (expres_nueva (nuevita $?A))
=>
  (assert (expresion $?A))
)

(defrule SIMPLIFICADOR::guarda_expression_para_exportar
  (declare (salience 2500))
  ?antes <- (expres $?A)
=>
  (open "c:\\algebra\\puente1.txt" puente "w")
  (printout puente $?A)
  (close puente)
  (retract ?antes)
)

(deffunction SIMPLIFICADOR::checa-exp ($?Var)
  (if (eq ^ (nth 1 $?Var))
    then (bind ?answer (nth 2 $?Var))
    else (bind ?answer 1))
  ?answer)

;*****
;          REGLA 1
;    {EXPRESION} ^ 0 => 1
;*****

(defrule SIMPLIFICADOR::exponente-cero-1
  (declare (salience 1500))
  ?antes <- (expresion $?A { $?B } ^ 0 $?C)
  (and
    (test (not (member { $?B})))
    (test (not (member } $?B))))
=>
  (assert (expres 1 $?A 1 $?C))
  (retract ?antes)
)

;*****
;          REGLA 2
;          A ^ 0 => 1
;*****

(defrule SIMPLIFICADOR::exponente-cero-2
  (declare (salience 1500))
  ?antes <- (expresion $?A ?B ^ 0 $?C)
  (test (neq ?B ))
=>
  (assert (expres 2 $?A 1 $?C))
  (retract ?antes)
```

```

)

;*****
;          REGLA 3
;  {EXPRESSION} ^ 1 => EXPRESION
;*****

(defrule SIMPLIFICADOR::exponente-uno-1
  (declare (salience 1400))
  ?antes <- (expresion $?A { $?B } ^ 1 $?C)
  (and
    (test (not (member { $?B}))
    (test (not (member } $?B))))
=>
  (assert (expres 3 $?A $?B $?C))
  (retract ?antes)
)

;*****
;          REGLA 4
;          A ^ 1 => A
;*****

(defrule SIMPLIFICADOR::exponente-uno-2
  (declare (salience 1400))
  ?antes <- (expresion $?A ?B ^ 1 $?C)
  (test (neq ?B ))
=>
  (assert (expres 4 $?A ?B $?C))
  (retract ?antes)
)

;*****
;          REGLA 5_A
;  {AB} ^ C => A^C * B^C
;*****

(defrule SIMPLIFICADOR::2da_Ley_Exponentes
  (declare (salience 1300))
  ?antes <- (expresion $?A { ?B $?C } ^ ?D $?E)
  (and
    (test (not (member { $?C}))
    (test (not (member } $?C))
    (test (not (member + $?C))
    (test (not (member - $?C))
    (test (not (member * $?C))
    (test (not (member ^ $?C))))
=>
  (assert (expresion $?A ?B ^ ?D { $?C } ^ ?D $?E))
  (retract ?antes)
)

;*****
;          REGLA 5_B
;  (A^X * B^Y)^C => A^(C*X) * B^(C*Y)
;*****

(defrule SIMPLIFICADOR::3ra_Ley_Exponentes
  (declare (salience 1300))
  ?antes <- (expresion $?A { $?B ?C ^ ?D $?E } ^ ?F $?G)
  (and
    (test (not (member { $?B}))

```

```

        (test (not (member ) $?B)))
        (test (not (member + $?B)))
        (test (not (member - $?B)))
        (test (not (member ( $?E)))
        (test (not (member } $?E)))
        (test (not (member + $?E)))
        (test (not (member - $?E)))
=>
  (assert (expression $?A ?C ^ =(* ?D ?F) { $?B $?E } ^ ?F $?G))
  (retract ?antes)
)

;*****
;          REGLA 5_C
;          () ^ C =>
;*****

(defrule SIMPLIFICADOR::Elimina_Perentesis
  (declare (salience 1300))
  ?antes <- (expresion $?A { } ^ ?B $?C)
=>
  (assert (expres 5 $?A $?C))
  (retract ?antes)
)

;*****
;          REGLA 6
;          2^3 => 8
;*****

;elevar una constante a un exponente.
(defrule SIMPLIFICADOR::constante-a-una-potencia
  (declare (salience 1200))
  ?antes <- (expresion $?A ?B&~x&~y&~z&~} ^ ?C $?D)
=>
  (assert (expres 6 $?A (integer (** ?B ?C)) $?D))
  (retract ?antes)
)

;*****
;          REGLA 7
;          4A*5B => 20A*B
;*****

(defrule SIMPLIFICADOR::simplifica-constantes-termino
  (declare (salience 1100))
;toma dos constantes y las almacena en ?B y ?D
  ?antes <- (expresion $?A ?B&~x&~y&~z&~{&~}&~+&~-&~*~&~^&~/ $?C
?D&~x&~y&~z&~{&~}&~+&~-&~*~&~^&~/ $?E)
; que la variable $?C no contenga => +, -, (, ni )
  (and
    (test (not (member + $?C)))
    (test (not (member - $?C)))
    (test (not (member / $?C)))
    (test (not (member { $?C)))
    (test (not (member } $?C))))
; que los ultimos miembros de $?A y $?C no sean ^
  (test (neq ^ =(nth (length $?A) $?A)))
  (test (neq ^ =(nth (length $?C) $?C)))
=>
  (assert (expres 7 $?A =(* ?B ?D) $?C $?E))
  (retract ?antes)
)

```

```

)

;*****
;           REGLA 8
;   A^B*A^C => A^(C+B)
;*****

(defrule SIMPLIFICADOR::1ra_Ley_Exponentes
  (declare (salience 1000))
  ?antes <- (expresion $?A ?B&x|y|z $?C ?D&x|y|z $?E)
  (test (eq ?B ?D))
;que la variable $?C no contenga => +, -, {, ni }
  (and
    (test (not (member + $?C)))
    (test (not (member - $?C)))
    (test (not (member / $?C)))
    (test (not (member { $?C)))
    (test (not (member } $?C))))
=>
  (bind ?Ex1 (checa-exp $?C))
  (bind ?Ex2 (checa-exp $?E))
  (if (neq ?Ex1 1)
    then
      (bind $?C (mv-delete 1 $?C))
      (bind $?C (mv-delete 1 $?C))
    )
  (if (neq ?Ex2 1)
    then
      (bind $?E (mv-delete 1 $?E))
      (bind $?E (mv-delete 1 $?E))
    )
  (assert (expres 8 $?A ?B ^ =(+ ?Ex1 ?Ex2) $?C $?E))
  (retract ?antes)
)

;*****
;           REGLA 9
;   A^B/ A^C => A^(B-C)
;*****

(defrule SIMPLIFICADOR::4ta_Ley_Exponentes
  (declare (salience 900))
  ?antes <- (expresion $?A ?B&x|y|z $?C / $?DD ?D&x|y|z $?E)
  (test (eq ?B ?D))
  (and
    (test (not (member + $?C)))
    (test (not (member - $?C)))
    (test (not (member / $?C)))
    (test (not (member { $?C)))
    (test (not (member } $?C))))
  (and
    (test (not (member + $?DD)))
    (test (not (member - $?DD)))
    (test (not (member / $?DD)))
    (test (not (member { $?DD)))
    (test (not (member } $?DD))))
=>
  (bind ?Ex1 (checa-exp $?C))
  (bind ?Ex2 (checa-exp $?E))
  (if (neq ?Ex1 1)
    then
      (bind $?C (mv-delete 1 $?C))

```



```

    (bind $?C (mv-delete 1 $?C))
  )
  (if (neq ?Ex2 1)
    then
      (bind $?E (mv-delete 1 $?E))
      (bind $?E (mv-delete 1 $?E))
    )
  (if (= ?Ex1 ?Ex2)
    then
      (assert (expression $?A 1 $?C / $?DD $?E))
    )
  (if (> ?Ex1 ?Ex2)
    then
      (assert (expression $?A ?B ^ =(- ?Ex1 ?Ex2) $?C / $?DD $?E))
    )
  (if (< ?Ex1 ?Ex2)
    then
      (assert (expression $?A $?C / $?DD ?B ^ =(- ?Ex2 ?Ex1) $?E))
    )
  (retract ?antes)
  (assert (bandera_control 1))
)

(defrule SIMPLIFICADOR::elimina-divisor-no-necesario-1
  (declare (salience 2000))
  ?antes <- (expression $?A / $?B)
;que la variable $?B contenga => +, - en el primer elemento
  (or (test (eq + (nth 1 $?B)))
      (test (eq - (nth 1 $?B)))
  )
=>
  (assert (expression $?A $?B))
  (retract ?antes)
)

(defrule SIMPLIFICADOR::elimina-divisor-no-necesario-2
  (declare (salience 2000))
  ?antes <- (expression $?A /)
=>
  (assert (expression $?A))
  (retract ?antes)
)

(defrule SIMPLIFICADOR::complementaria
  (declare (salience 800))
  ?antes1 <- (expression $?A)
  ?antes2 <- (bandera_control 1)
=>
  (assert (expres 9 $?A))
  (retract ?antes1)
  (retract ?antes2)
)

;*****
;          REGLA 10
;      Y^3*X^2*Z => X^2*Y^3*Z
;*****

(defrule SIMPLIFICADOR::reacomoda-variables-termino
  (declare (salience 600))
  (or
    ?antes <- (expression $?A ?B&y|z $?C ?D&x $?E)
  )
)

```

```

    ?antes <- (expresion $?A ?B&z $?C ?D&y $?E))
;que la variable $?C no contenga => +, -, {, ni }
  (and
    (test (not (member + $?C)))
    (test (not (member - $?C)))
    (test (not (member / $?C)))
    (test (not (member { $?C)))
    (test (not (member } $?C)))
  )
=>
  (bind ?Ex1 (checa-exp $?C))
  (bind ?Ex2 (checa-exp $?E))
  (if (and (neq ?Ex1 1) (neq ?Ex2 1))
    then
      (bind $?C (mv-replace 2 $?C ?Ex2))
      (bind $?E (mv-replace 2 $?E ?Ex1))
    else
      (if (neq ?Ex2 1)
        then
          (bind $?E (mv-delete 1 $?E))
          (bind $?E (mv-delete 1 $?E))
          (bind $?C (mv-append $?C ^ ?Ex2))
        )
      (if (neq ?Ex1 1)
        then
          (bind $?C (mv-delete 1 $?C))
          (bind $?C (mv-delete 1 $?C))
          (bind $?E (mv-append ^ ?Ex1 $?E))
        )
      )
  )
  (assert (expres 10 $?A ?D $?C ?B $?E))
  (retract ?antes)
)

;*****
;          REGLA 11
;    X*Y^2*8Z => 8X*Y^2*Z
;*****

(defrule SIMPLIFICADOR::reacomoda-constante-termino
  (declare (salience 500))
  ?antes <- (expresion $?A $?C ?D&~x&~y&~z&~{&~}&~+&~-&~*&~^&~/ $?E)
;que la variable $?C no contenga => +, -, {, ni }
  (and
    (test (not (member + $?C)))
    (test (not (member - $?C)))
    (test (not (member / $?C)))
    (test (not (member { $?C)))
    (test (not (member } $?C)))
  )
  (test (> (length $?C) 0))
  (test (neq (nth (length $?C) $?C) ^))
  (or (test (member (nth (length $?A) $?A) (create$ / - + } {})))
      (test (= (length $?A) 0))
  )
=>
  (assert (expres 11 $?A ?D $?C $?E))
  (retract ?antes)
)

;*****
;          REGLA 12

```

```

;      4Y^3 + 2Y^3 - Y^3 => 5Y^3
;*****

(defrule SIMPLIFICADOR::Suma_Algebraica_1 "separa terminos"
  (declare (salience 400))
  ?antes <- (expression $?A ?B*+|- $?G)
  (test (> (length $?A) 0))
  (and
    (test (not (member { $?A})))
    (test (not (member } $?A)))
    (test (not (member + (rest$ $?A))))
    (test (not (member - (rest$ $?A))))
  )
=>
  (if (and (not (member - $?G)) (not (member + $?G)))
    then
      (if (not (integerp (nth 1 $?G))) then (assert (termino ?B 1 $?G =(gensym)))
        else (assert (termino ?B $?G =(gensym)))
      )
      (assert (uniendo))
      (assert (permiso r12))
    else
      (assert (expression ?B $?G))
    )
  (if (or (eq + (nth 1 $?A))
    (eq - (nth 1 $?A))
  )
  then
    (if (not (integerp (nth 2 $?A)))
      then
        (bind ?signo (first$ $?A))
        (bind $?resto (rest$ $?A))
        (assert (termino ?signo 1 $?resto =(gensym)))
        else (assert (termino $?A =(gensym)))
      )
    else
      (if (not (integerp (nth 1 $?A)))
        then (assert (termino + 1 $?A =(gensym)))
        else (assert (termino + $?A =(gensym)))
      )
    )
  )
  (retract ?antes)
)

;sumar terminos semejantes
(defrule SIMPLIFICADOR::Suma_Algebraica_2 "suma terminos semejantes"
  (declare (salience 700))
  ?antes1 <- (termino ?A ?B ?C ?IDE1)
  ?antes2 <- (termino ?D ?E ?C ?IDE2)
  (test (neq ?IDE1 ?IDE2))
  (permiso r12)
=>
  (if (eq ?A ?D)
    then
      (assert (termino ?A =(+ ?B ?E) ?C ?IDE1))
    else
      (if (<> ?B ?E)
        then
          (if (> ?B ?E)
            then
              (assert (termino ?A =(- ?B ?E) ?C ?IDE1))
            else

```

```

        (assert (termino ?D =(- ?E ?B) ?C ?IDE1))
    )
    else (assert (termino 0))
    )
)
(retract ?antes1)
(retract ?antes2)
)

(defrule SIMPLIFICADOR::Suma_Algebraica_3 "une terminos"
  (declare (salience 500))
  ?antes1 <- (uniendo $?A)
  ?antes2 <- (termino ?B ?BB ?C ?IDE2)
  (permiso r12)
=>
  (if (and (eq + ?B) (= 0 (length $?A)))
    then
      (if (> ?BB 1)
        then (assert (uniendo ?BB ?C))
        else (assert (uniendo ?C))
      )
    else
      (if (> ?BB 1)
        then (assert (uniendo $?A ?B ?BB ?C))
        else (assert (uniendo $?A ?B ?C))
      )
    )
  (retract ?antes1)
  (retract ?antes2)
)

(defrule SIMPLIFICADOR::Suma_Algebraica_4 "convierte union"
  (declare (salience 300))
  ?antes1 <- (uniendo $?A)
  ?antes2 <- (permiso r12)
=>
  (if (> (length $?A) 0) then (assert (expres 12 $?A))
  else (assert (expres 12 0))
  )
  (retract ?antes1)
  (retract ?antes2)
)

```

## Anexo 2. Relación de Consejos según la regla errónea aplicada.

Regla incorrecta 1	<p><math>A^0 \Rightarrow 0</math> Incorrecto  <math>A^0 = 1</math> Correcto</p> <p>Explicación:</p> <p>El exponente cero proviene de la división de potencias de la misma base:</p> $A^2 / A^2 = 1 = A^{(2-2)} = A^0 = 1$ <p>Toda cantidad elevada a la cero equivale a 1, ya que toda cantidad dividida por si misma es igual a 1.</p>
Regla incorrecta 2	<p><math>A^0 \Rightarrow A</math> Incorrecto  <math>A^0 = 1</math> Correcto</p> <p>Explicación:</p> <p>El exponente cero proviene de la división de potencias de la misma base:</p> $A^2 / A^2 = 1 = A^{(2-2)} = A^0 = 1$ <p>Toda cantidad elevada a la cero equivale a 1, ya que toda cantidad dividida por si misma es igual a 1.</p>
Regla incorrecta 3	<p><math>A^1 \Rightarrow 1</math> Incorrecto  <math>A^1 = A</math> Correcto</p> <p>Explicación:</p> <p>El exponente indica las veces que la base es tomada como factor, así:</p> $A^2 = A * A$ $A^3 = A * A * A$ $A^1 = A$ <p>Toda cantidad elevada al exponente 1 es igual a la misma cantidad.</p>
Regla incorrecta 4	<p><math>(A+B)^C \Rightarrow A^C + B^C</math> Incorrecto  <math>(A+B)^C = (A+B)^C</math> Correcto</p> <p>Explicación:</p> <p>Esta expresión es un binomio a una potencia y no puede simplificarse usando las Leyes de los Exponentes. Puede ser que estés confundido con la aplicación de la segunda Ley de los Exponentes:</p> $(A*B)^C = A^C + B^C$ <p>Fíjate que para este caso la expresión tiene solamente factores dentro del paréntesis.</p>

Regla incorrecta 5	$(A^B)^C \Rightarrow A^{(B \cdot C)}$ Incorrecta $(A^B)^C = A^{(B^C)}$ Correcto  Explicación:  El exponente indica las veces que la base es tomada como factor, así:  $(A^2)^3 = (A^2) \cdot (A^2) \cdot (A^2) = (A \cdot A) \cdot (A \cdot A) \cdot (A \cdot A) = A \cdot A \cdot A \cdot A \cdot A \cdot A = A^6$  Segunda Ley: La potencia de una potencia es igual a la base elevada a multiplicación de los exponentes
Regla incorrecta 6	$A \cdot 0 \Rightarrow A$ Incorrecto $A \cdot 0 = 0$ Correcto  Explicación:  La constante 0 indica las veces que será sumada la cantidad que esta multiplicando. $A \cdot 3 = A + A + A$ $A \cdot 1 = A$ $A \cdot 0 = 0$  Toda cantidad multiplicado por cero es igual a cero.
Regla incorrecta 7	$A \cdot 1 \Rightarrow 1$ Incorrecto $A \cdot 1 = A$ Correcta  Explicación:  La constante 1 indica las veces que será sumada la cantidad que esta multiplicando.  $A \cdot 3 = A + A + A$ $A \cdot 1 = A$  Toda cantidad multiplicada por 1 es igual a la misma cantidad.
Regla incorrecta 8	$3 \times 4y \Rightarrow 7xy$ Incorrecto $3 \times 4y = 12xy$ Correcto  Explicación:  Las constantes dentro de un término de una expresión, se multiplican. Dentro de un término todas las variables y constantes son factores.

Regla incorrecta 9	$2xyx \Rightarrow 3xy$ Incorrecto $2xyx = 2x^2y$ Correcto  Explicación:  Al manipular expresiones algebraica debemos recordar que todas las constantes y variables dentro de cada término representan factores, es decir, se están multiplicando.  Por lo tanto, las constantes sólo se pueden multiplicar con otras constantes dentro del mismo término.
Regla incorrecta 10	$A^B * A^C \Rightarrow A^{(B*C)}$ Incorrecto $A^B * A^C = A^{(B+C)}$ Correcto  Explicación:  En este caso debe aplicarse la Segunda Ley de los Exponentes.  $A^2 * A^3 = A * A * A * A * A = A^5 = A^{(2+3)}$  Cuando se multiplican dos potencias de la misma base, los exponentes se suman.
Regla incorrecta 11	$A^B / A^C \Rightarrow A^{(B/C)}$ Incorrecto $A^B / A^C = A^{(B-C)}$ Correcto  Explicación:  En este caso debe aplicarse la Cuarta Ley de los Exponentes.  $A^3 / A^2 = A * A * A / A * A = A = A^{(3-2)} = A^1 = A$  Cuando se dividen dos potencias de la misma base, el exponente de denominador se resta del exponente del denominador.
Regla incorrecta 12	$A / A \Rightarrow 0$ Incorrecto $A / A = 1$ Correcto  Explicación:  <u>Toda cantidad dividida por si misma es igual a 1.</u>
Regla incorrecta 13	$A^B + A^C \Rightarrow A^{(B+C)}$ Incorrecto $A^B + A^C = A^B + A^C$ Correcto  Explicación:  No existe una Ley de los Exponentes que me permita simplificar una expresión de este tipo. Posiblemente estés confundiendo la Primera Ley de los Exponentes. La Primera Ley de los Exponentes:  $A^B * A^C \Rightarrow A^{(B+C)}$  sólo se aplica cuando las variables pertenecen a un mismo término, es decir, cuando NO existen signos de suma o resta entre ellos.

### Anexo 3. Cuestionarios aplicados para el diseño de interfaz.

#### CUESTIONARIO PARA EL ALUMNO

1.- ¿Cuántas veces a la semana practicas deporte?

 7 3 1 menos de una

2.- ¿Si padeces alguna enfermedad de la vista, especifica cuál (miopía, astigmatismo, daltonismo, etc.)?

\_\_\_\_\_

3.- ¿Si padeces alguna enfermedad crónica, especifica cuál?

\_\_\_\_\_

4.- ¿Cómo consideras el estado de tu sentido del oído?

 excelente bueno regular malo

5.- ¿Con quien prefieres practicar tus ejercicios de álgebra?

 solo con el maestro con compañeros

6.- ¿Cómo prefieres resolver tus dudas al practicar tus ejercicios de álgebra?

 En libros preguntado al maestro preguntando a compañeros

7.- Relaciona cada concepto con algún símbolo, escribiendo la letra del símbolo que mejor representa el texto: (Si es otro especifica cuál)

( ) Objeto donde resuelvo mis ejercicios de álgebra.

( ) Agregar algo nuevo o proponer algo nuevo cuando resuelvo un ejercicio.

( ) Arrepentirse de una respuesta cuando resuelvo un ejercicio.

( ) Puede aclarar mis dudas.

a.- libreta

b.- pizarrón

c.- hoja de papel

d.- 


e.- gis

f.- 

g.- borrador

h.- 

j.- 

k.- 

l.- 

m.- 

n.- 

o.- 

p.- maestro



8.- ¿Dónde prefieres practicar tus ejercicios de álgebra?

salón de clases       Biblioteca       casa

otro lugar : \_\_\_\_\_

9.- ¿Usas con regularidad:  
(Puedes contestar más de una)

computadora?       el microsoft Word?       el Netscape?  
 el Power Point?       algún programa gráfico?

10.- ¿Cuáles son tus pasatiempos preferidos?

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

11.- ¿Marca las condiciones ambientales de la sala de cómputo?

Iluminación:

excelente       Regular       mala

Ventilación y aire acondicionado:

excelente       Regular       mala

Ruido:

alto       Medio       casi silencio

12.- ¿Que tan seguido juegas con videojuegos?

muy seguido       Regularmente       poco  
 nada

## CUESTIONARIO PARA EL MAESTRO

1.- ¿Cuántas veces a la semana practicas deporte?

- 7                       3                       1                       menos de una

2.- ¿Si padeces alguna enfermedad de la vista, especifica cuál (miopía, astigmatismo, daltonismo, etc.)?

\_\_\_\_\_

3.- ¿Si padeces alguna enfermedad crónica, especifica cuál?

\_\_\_\_\_

4.- ¿Cómo consideras el estado de tu sentido del oído?

- excelente               bueno               regular               malo

5.- ¿Usas con regularidad:  
(Puedes contestar más de una)

- computadora?               el microsoft Word?               el Netscape?  
 el Power Point?               algún otro programa gráfico?

6.-¿Qué consideras que es más importante conocer de un alumno al enseñar, su conocimiento erróneo o su conocimiento correcto?

7.-¿Cuál es tu forma de trabajo para detectar las causas de los errores de los alumnos al resolver problemas de álgebra?

8.-¿Cual es la mejor manera de describir una error en álgebra (en forma NO verbal) de un alumno? Describe un error:

9.- ¿Cuáles son tus pasatiempos preferidos?

---

---

---

10.- ¿Marca las condiciones ambientales de tu área de trabajo?

Iluminación:

excelente                       regular                       mala

Ventilación y aire acondicionado:

excelente                       regular                       mala

Ruido:

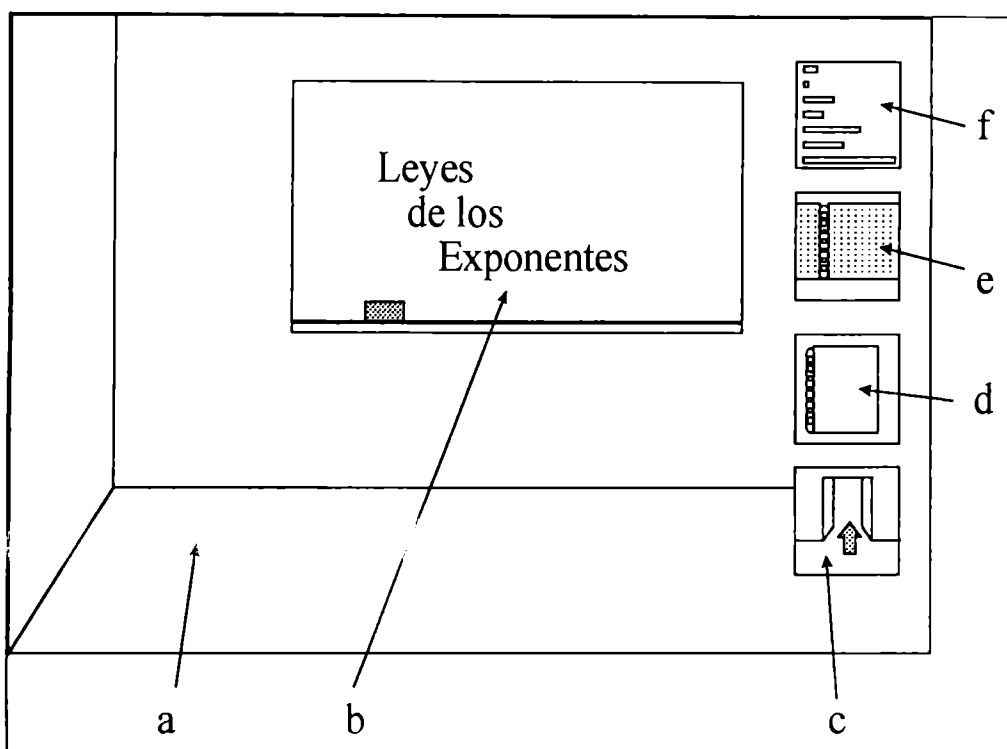
alto                                       medio                                       casi silencio

## Anexo 4. Pantallas.

The diagram shows a user verification interface. It features a header bar at the top. Below it, there are four input fields: 'Matrícula:', 'Nombre:', 'Apellidos:', and 'Usuario:'. The 'Usuario:' field is a dropdown menu with 'Alumno' and 'Maestro' as options. To the right of these fields is a decorative image of a door. At the bottom of the screen, there is a button with a door icon and an upward-pointing arrow. Labels 'a' through 'f' are placed around the interface with arrows pointing to specific elements: 'a' points to the 'Usuario:' dropdown, 'b' points to the decorative wall area, 'c' points to the door icon button, 'd' points to the 'Matrícula:' input field, 'e' points to the 'Nombre:' input field, and 'f' points to the 'Apellidos:' input field.

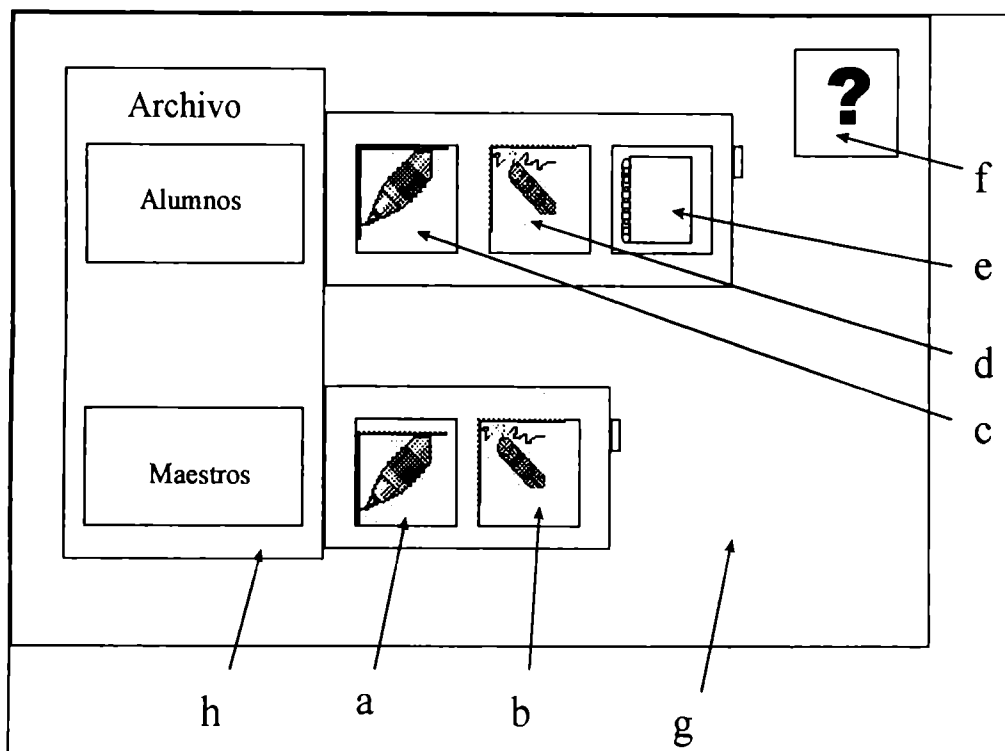
Pantalla para verificar identidad y tipo de usuario:

- Elemento de selección. Se selecciona una de las opciones: alumno o maestro.
- Elemento decorativo, representa la pared de la fachada de una escuela.
- Icono que representa una puerta con una flecha que indica que se atravesará. Permite iniciar la sesión cuando se han capturado todos los datos del usuario.
- Elemento de captura. Captura la matrícula del usuario.
- Elemento de captura. Captura el nombre del usuario.
- Elemento de captura. Captura los apellidos del usuario.



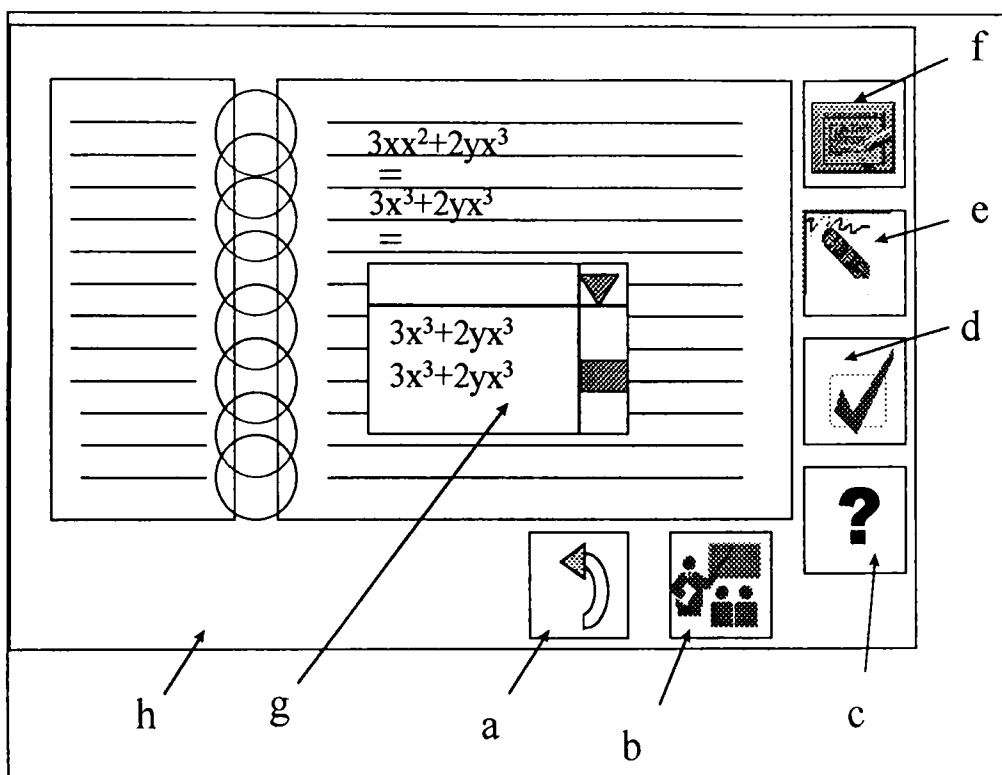
Pantalla principal del alumno:

- a. Símbolo de ambiente escolar, representa un aula. Cumple una función decorativa.
- b. Símbolo de ambiente escolar, representa un pizarrón. Anuncia el tema.
- c. Icono, representa un puerta con una flecha iniciando que se va atravesar la puerta. Permite salir del sistema.
- d. Icono que representa una libreta cerrada. Permite revisar la historia de errores.
- e. Icono, representa una libreta abierta. Permite iniciar la sesión de ejercicios.
- f. Icono que representa un gráfico de barras. Permite revisar el modelo de conocimientos del estudiante.



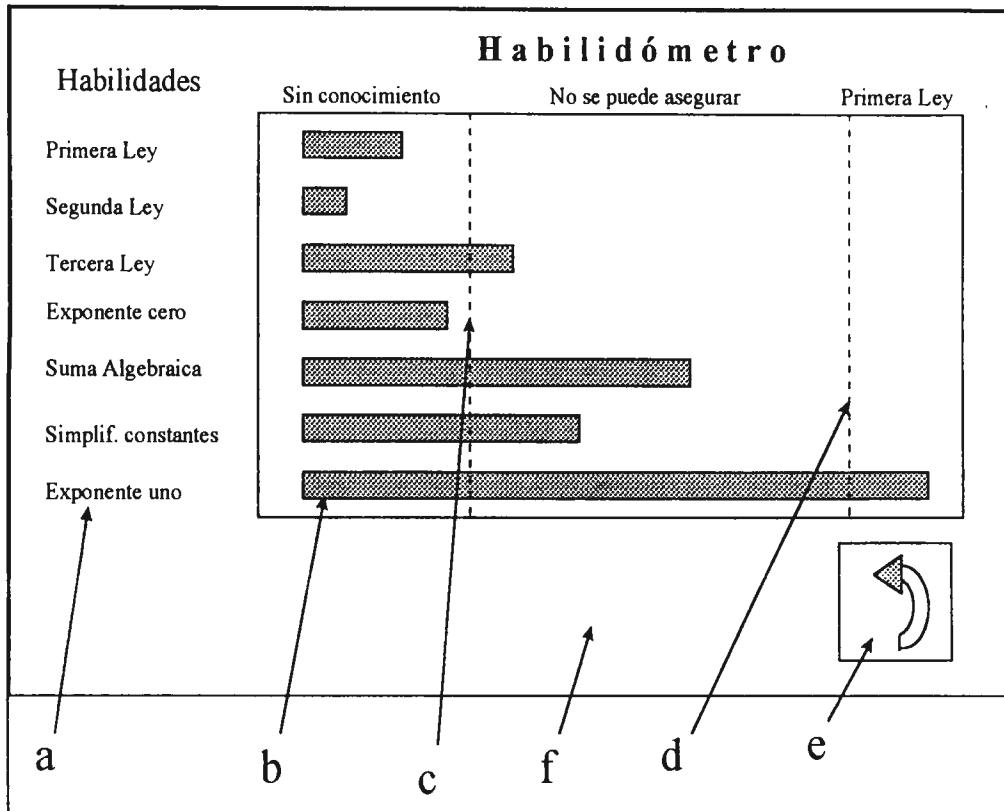
Pantalla principal del maestro:

- a. Icono que representa una pluma en posición de escritura. Permite dar de alta a un maestro en el sistema.
- b. Icono que representa un lápiz borrando. Permite dar de baja a un maestro del sistema.
- c. Icono que representa una pluma en posición de escritura. Permite dar de alta a un alumno en el sistema.
- d. Icono que representa un lápiz borrando. Permite dar de baja a un alumno del sistema.
- e. Icono que representa una libreta cerrada. Permite revisar la historia de errores cometidos por un estudiante.
- f. Icono que representa un signo de interrogación. Permite al usuario solicitar ayuda en algún tópico relacionado con el uso del sistema.
- g. Fondo decorativo que representa la oficina del maestro.
- h. Elemento que nos dice que la pantalla actual nos facilita el acceso a la información almacenada por el sistema.



Pantalla para la sesión de ejercicios:

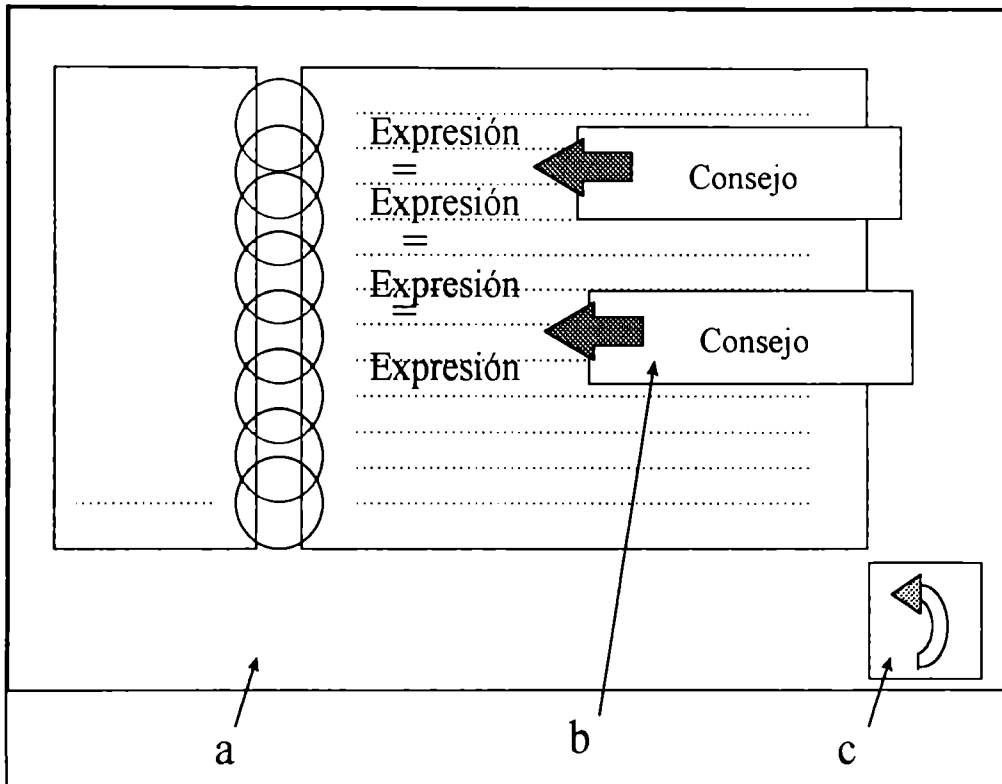
- a. Icono que representa una flecha en U. Permite volver a la pantalla principal del alumno.
- b. Icono que representa a un maestro impartiendo cátedra. Permite al estudiante consultar la teoría relacionada con el tema de la Leyes de los Exponentes.
- c. Icono que representa un signo de interrogación. Permite al usuario consultar información sobre el manejo del sistema.
- d. Icono que representa una palomita. Permite al usuario decirle al sistema que el ejercicio ha sido terminado.
- e. Icono que representa un lápiz borrando. Permite al usuario borrar la última entrada.
- f. Icono que representa un lápiz escribiendo. Permite escribir una expresión una vez seleccionada.
- g. Elemento de selección. Permite la selección de la siguiente expresión durante la solución de un ejercicio.
- h. Elemento decorativo que representa una libreta sobre una mesa de madera.



Pantalla para presentar el modelo de conocimientos del estudiante:

- Elemento que representa una Ley algebraica.
- Gráfica de barra que representa la magnitud del conocimiento del estudiante en una Ley del álgebra específica.
- División para las leyes conocidas. Si la gráfica NO pasa esa división, entonces el estudiante NO conoce esa ley.
- División para las leyes conocidas. Si la gráfica pasa esa división, entonces el estudiante SI conoce esa ley.
- Icono que representa una flecha en U. Permite al usuario volver a la pantalla principal del alumno o del maestro según sea el caso.
- Elemento decorativo.





Pantalla para presentar la historia de errores y los consejos al estudiante:

- Elemento decorativo que representa un a libreta sobre una mesa de madera.
- Región donde se muestra el consejo al estudiante. La flecha señala el lugar donde se cometió el error.
- Icono que representa una flecha en U. Permite al usuario volver a la pantalla anterior.

Estudiante a dar de alta:

Matrícula:

Nombre:

Apellidos:

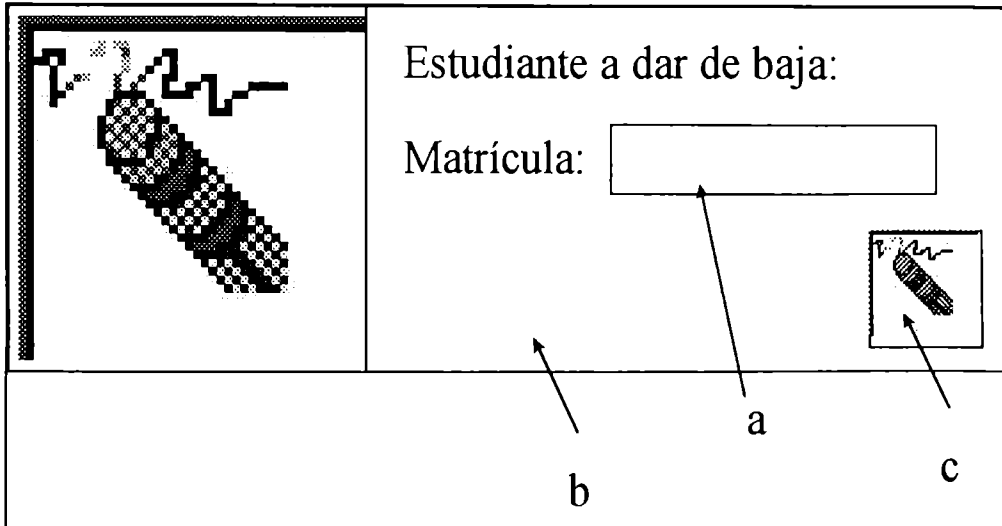
a

b

c

Pantalla para dar de alta a alumnos y maestros:

- a. Elemento decorativo.
- b. Elemento de captura. Permite capturar los datos del alumno o maestro a dar de alta.
- c. Icono que representa una pluma escribiendo. Permite dar de alta al usuario cuando se han llenado todos los datos.



Pantalla para dar de baja a alumnos y maestros:

- a. Elemento decorativo.
- b. Elemento de captura. Permite capturar la matrícula del usuario a dar de baja.
- c. Icono que representa un lápiz borrando. Permite ejecutar la acción de borrar una vez que se ha capturado la matrícula del usuario.

## Bibliografía.

ACT Research Group, *The Practical Algebra Tutor (PAT)*, <http://act.psy.cmu.edu/ACT/awpt/awpt-home.html>, 1997, (8/Nov./1997).

ACT Research Group, *PAT on line*, <http://act.psy.cmu.edu/pat-online.html>, 1997, (8/Nov./1997).

Anderson, John R., The Expert Module, Polson, Martha C y J. Jeffrey Richardson, editores, *Foundations of Intelligent Tutoring Systems*, págs. 21-53, New Jersey, 1988, Lawrence Erlbaum Associates Publisher.

Anderson, John, Intelligent Tutoring and High School Mathematics, C. Frasson, G. Gauthier, G. I. McCalla, Editores, *Intelligent Tutoring Systems: Second International Conference ITS'92*, Proceedings, págs. 1-20, Montreal, Canada, 1992. Springer-Verlag.

Anthena Information Service Inc., *CLIPS Dynamic Link Library for Windows 95/NT*, <http://ourworld.compuserve.com/homepages/marktoml/clipstuf.htm>, 1997, (1/Dic./1997).

Bachelard, Gastón, *La formación del espíritu científico*, págs. 15-26, México, 1988, Siglo veintiuno editores.

Brown, J. S., y VanLehn, K. (1980). *Repair theory: A generative theory of bugs in procedural skills*. *Cognitive Science*, 4, 379-426.

Burns, Hugh L. y Charles G. Capps, *Foundations of Intelligent Tutoring Systems: An Introduction*, Polson, Martha C y J. Jeffrey Richardson, editores, *Foundations of Intelligent Tutoring Systems*, págs. 1-19, New Jersey, 1988, Lawrence Erlbaum Associates Publisher.

Burton, R. R., y Brown, J.S.(1982). An investigation of computer coaching for informal learning activities. In D. Sleeman y J. S. Brown (Eds), *Intelligent tutoring systems* (pp. 79-98). New York: Academic Press.

Burton, Richard R., The Environment Module of Intelligent Tutoring Systems, Polson, Martha C y J. Jeffrey Richardson, editores, *Foundations of Intelligent Tutoring Systems*, pags. 109-142, New Jersey, 1988, Lawrence Erlbaum Associates Publisher.

Campbell Dávila, Moraima, *Sistemas de información multimedia: material de apoyo de la materia*, ITESM, Campus Monterrey, 1997.

Carbonell, J. R. (1970). *AI in CAI: An artificial intelligent approach to computer-aided instruction*. IEEE Transactions on Man-Machine Systems, 11, 190-202.

Clancey, W. J. (1982). Tutoring rules for guiding a case method dialogue. In D. Sleeman y J. S. Brown (Eds), *Intelligent tutoring systems* (pp. 201-225). New York: Academic Press.

Chambers, Jack A. y Sprecher, Jerry W., *Computer-Assisted Instruction: Its Use in the Classroom*, pags. 3-25, New Jersey, 1983, Prentice-Hall Inc.

Friedman-Hill, Ernest, *JESS, The Java Expert System Shell*, <http://herzberg.ca.sandia.gov/jess/>, 1997, (27/Nov./1997).

Gentner, Donald R., and Grudin, Jonathan, *Design Models for Computer-Human Interaction*, Computer, Vol. 29, No. 6, Junio 1996, pp 28-35.

Giordano, Mike, *WebCLIPS Demo Page*, <http://www.monmouth.com/~km2580/wchome.htm>, 1997, (27/Nov./1997).

Gisolfi, Antonio, Dattolo, Antonina y Balzano, Walter, *A fuzzy approach to student modeling*, 1992.

Hagen Estrada, José Alessio, *Metáforas de Diseño de una Interfase y su Rol dentro de la Interacción Hombre Máquina*, Reporte de Investigación, Curso de Diseño de Sistemas Interactivos, Semestre enero-mayo de 1995, ITESM Campus Monterrey, abril de 1995.

Half, Henry M., Curriculum and Instruction in Automated Tutors, Polson, Martha C y J. Jeffrey Richardson, editores, *Foundations of Intelligent Tutoring Systems*, pags. 79-108, New Jersey, 1988, Lawrence Erlbaum Associates Publisher.

Janlert, Lars-Erik, *Models in Human-Computer Interaction*, Research Report UMINF-161.89, ISSN 0348-05442, Information Processing Institute, University of Umea, Suecia, 1989.

Johnsom, L., y Soloway, E. (1984). *PROUST: Knowledge-based program debugging*, Proceedings of the Seventh International Software Engineering Conference (pp. 369-380).

Kilpatrick, Jeremy, Luis, Investigación en Educación matemática: su historia y algunos temas de actualidad, *Educación matemática*, Editores Jeremy Kilpatrick y otros, México, 1993, pags. 1-18, Grupo Editorial Iberoamérica.

Marcus, Aaron, *The Cross-GUI Handbook: For Multiplatform User Interface Design*, pags. 1-14, USA, 1995, Addison-Wesley Publishing Company.

Matz, M., Toward a process model for high school algebra errors, *Intelligent Tutoring Systems*, Serie: Computer and People, Sleeman, D. y Brown, J. S., pags. 25-50, 1982, Editores, Academic Press.

Miller, James R., The Role of Human-Computer Interaction In Intelligent Tutoring Systems, Polson, Martha C y J.Jeffrey Richardson, editores, *Foundations of Intelligent Tutoring Systems*, pags. 143-189, New Jersey, 1988, Lawrence Erlbaum Associates Publisher.

Miller, M. L. (1982). A structured planning environment for elementary programming. In D. Sleeman y J. S. Brown (Eds), *Intelligent tutoring systems* (pp. 119-135). New York: Academic Press.

Reiser, B. V., Anderson, J. R., y Farrell, R. G. (1985). *Dynamic student modeling in an intelligent tutor for lisp programming*. Proceedings of Ninth International Joint Conference on Artificial Intelligent (pp. 8-14). Los Altos, CA: Morgan Kaufman.

Rico, Luis, Errores y dificultades en el aprendizaje de las matemáticas, *Educación matemática*, Editores Jeremy Kilpatrick y otros, México, 1993, pags. 69-108, Grupo Editorial Iberoamérica.

Riley, Gary, *CLIPS: A Tool for Building Expert Systems*, <http://www.jsc.nasa.gov/~clips/CLIPS.html>, 1997, (27/Nov./1997).

Shortliffe, E. H. (1976). *Computer-based medical consultations: MYCIN*. New York: American Elsevier.

Toro Palacios, Margarita, *Aplicación de la metodología de procesos en el diseño y desarrollo de materiales instruccionales del área de las matemáticas a nivel preparatoria*, Tesis de Maestría, ITESM Campus Eugenio Garza Sada, 1996.

Sun Microsystems Inc., *Java Home Page*, <http://www.javasoft.com/>, 1997, (27/Nov./1997).

VanLehn, Kurt, Student Modelling, Polson, Martha C y J.Jeffrey Richardson, editores, *Foundations of Intelligent Tutoring Systems*, pags. 55-78, New Jersey, 1988, Lawrence Erlbaum Associates Publisher.

## **Vita.**

Benjamín Eduardo García Villegas nació en Culiacán, Sinaloa, México, el 20 de Noviembre de 1967.

Recibió el grado de Ingeniero Bioquímico en la Universidad Autónoma de Sinaloa, en la ciudad de Culiacán, Sinaloa en Junio de 1991.

Se ha desempeñado en el área académica y de la docencia, y prestó sus servicios al Centro de Ciencias de Sinaloa durante dos años en uno de sus laboratorios.

El mes de Enero de 1996 ingresó al Programa de Graduados en Informática del Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Monterrey.

**Dirección Permanente**  
Bahía de Agiabampo #856,  
Fracc. Insurgentes, C.P. 80026,  
Culiacán, Sinaloa, México.  
Tel. (0167) 13-16-35

La presente tesis fue mecanografiada por el autor, usando las facilidades computacionales del ITESM.