

# ***Anagra 2.0. Un entorno para el estudio de las fases de análisis en el desarrollo de traductores***

## **Manual de usuario**

Realizado por: **Raúl Novoa Mínguez**

Dirigido por: **Joaquín Ezpeleta Mateo**



Dpto. Informática e Ingeniería de  
Sistemas



Universidad de Zaragoza

# Manual de Usuario

## 1. Introducción

### 1.1. Guía de lectura del manual

La ejecución de ANAGRA así como el uso de este manual, está dirigido a usuarios habituales de sistemas de ventanas (Windows, X-Windows,...). Los usuarios de ANAGRA son personas con conocimientos avanzados de informática, por lo que la descripción de las opciones es bastante técnica. Esta aplicación realiza el análisis de gramáticas para determinar su tipo, por lo que es conveniente un conocimiento general sobre este tema.

Asimismo, el programa permite la simulación de los analizadores sintácticos asociados a las gramáticas con el fin de mostrar paso a paso como funcionan éstos. Se incluyen también métodos para la transformación de gramáticas en otras equivalentes para conseguir que las gramáticas sean de un tipo determinado.

Es conveniente la lectura de la sección de procedimientos, con el fin de realizar las tareas de esta aplicación, en caso de duda, con mayor seguridad. En caso de ejecución errónea de estas opciones, existe una sección de análisis de errores, donde se proporciona al usuario una relación de los errores más frecuentes, su posible causa y su solución.

### 1.2. Propósito

El propósito del documento actual es ofrecer una ayuda al futuro usuario de la aplicación. En este documento se enunciarán todas las posibilidades que ofrece la aplicación así como los pasos que hay que ejecutar para poder llevarlas a cabo. De igual manera se intentarán solucionar todos los posibles problemas que se plantean al usuario mediante el manejo e instalación del programa.

El propósito del software desarrollado es poder analizar gramáticas, de manera que el usuario pueda reconocer si la gramática que ha introducido al programa es del tipo LL(1), SLR(1), LR(1) ó LALR(1). Para cada uno de estos análisis el programa mostrará todos los pasos ejecutados así como los resultados intermedios que se obtienen en la comprobación del tipo de gramática (tablas...), para que el usuario no sólo sepa si una gramática es de un tipo u otro, sino que además pueda deducir por qué la gramática no es de un tipo determinado.

Por otra parte una vez que determinemos de qué tipo es la gramática introducida podremos simular los distintos analizadores sintácticos asociados a dichas gramáticas, introduciéndoles una entrada de símbolos y comprobando si dicha entrada es reconocida o no por la gramática. En este caso tendremos dos opciones, una de ellas será ver todo el proceso de análisis de la entrada viendo todas las estructuras internas que se utilizan en el análisis. La segunda opción de simulación será la introducción directa del conjunto de símbolos a analizar, y el programa nos dirá directamente si dicha entrada es reconocida o no por la gramática pero sin entrar en ningún tipo de detalles de porque es así. Para ambas opciones podremos obtener el árbol de sintaxis resultante de la simulación.

Además el programa nos da la posibilidad de operar sobre las gramáticas editándolas y aplicando sobre ellas distintas transformaciones (que se verán mas adelante) de modo que generen otras gramáticas equivalentes que se adapten así a un tipo concreto de gramática.

También nos ofrece el programa los conjuntos PRI y SIG de los símbolos de la gramática y obtener el conjunto PRI de una determinada forma de frase formada por símbolos de la gramática.

### 1.3. Visión general del documento

Este manual comprende unas secciones introductorias acerca de la herramienta y sus funcionalidades (sección 1).

En la sección 2 podemos encontrar la descripción funcional del software en sí, esto es qué hace cada una de las opciones que podremos elegir al ejecutar nuestra aplicación, así como precauciones que debemos tomar al ejecutarlas y una lista de errores, posibles causas y comportamientos de la aplicación si estos sucedieran. También se hace una referencia a la descripción de entradas y resultados para cada una de las operaciones implementadas.

En la sección 3 encontramos un tutorial del sistema que describe la manera de poder realizar todos los procedimientos descritos en la sección 2.

## 2. Descripción funcional del software

### 2.1. Procedimientos y precauciones

#### 2.1.1. Apertura de la gramática

##### Previos

- Aplicación inicializada con éxito.
- Aparición de un diálogo de apertura de fichero donde seleccionaremos el fichero de la gramática que queremos abrir.

##### Resultados tras la apertura de la gramática

- **Ejecución exitosa:**  
Gramática mostrada en pantalla, y calculados sus conjuntos PRI y SIG.
- **Ejecución errónea:**  
Ver errores de apertura de gramática.

#### 2.1.2. Edición de la gramática

##### Previos

- Aplicación inicializada con éxito (si se desea escribir la gramática).
- Apertura correcta de la gramática (si se desea modificar una gramática ya abierta).

##### Resultados tras la edición de la gramática

- **Ejecución exitosa:**  
Gramática editada mostrada en pantalla, y calculados sus conjuntos PRI y SIG.
- **Ejecución errónea:**  
Ver errores de edición de gramáticas.

### **2.1.3. Cálculo del conjunto PRI de una forma de frase**

#### **Previos**

- Apertura o edición correcta de la gramática.
- Aparición de la ventana donde se introduce la forma de frase. Introducimos la forma de frase que queremos analizar.

#### **Resultados tras el cálculo del conjunto PRI de una forma de frase**

- **Ejecución exitosa:**  
Muestra en pantalla del conjunto de símbolos PRI correspondiente a la forma de frase que hemos introducido.
- **Ejecución errónea:**  
Ver errores del cálculo del conjunto PRI de una forma de frase.

### **2.1.4. Transformaciones de las gramáticas**

#### **Previos**

- Apertura o edición correcta de la gramática.
- Selección del menú del tipo de transformación que queremos aplicar.

#### **Resultados tras el cálculo del conjunto PRI de una forma de frase**

- **Ejecución exitosa:**  
Gramática equivalente mostrada en una ventana aparte. También quedan calculados los conjuntos PRI y SIG de la nueva gramática.
- **Ejecución errónea:**  
Ver errores de transformación de gramáticas.

### **2.1.5. Análisis LL(1)**

#### **Previos**

- Apertura o edición correcta de la gramática.

#### **Resultados tras el análisis LL(1)**

- **Ejecución exitosa:**  
Se muestra en pantalla la tabla del análisis LL(1) correspondiente a dicha gramática. En esta tabla se especifica si la gramática es o no del tipo LL(1) y las causas.
- **Ejecución errónea:**  
Ver errores del análisis LL(1).

### **2.1.6. Ver tabla de análisis LL(1)**

#### **Previos**

- Gramática analizada según el tipo LL(1).

#### Resultados tras ver la tabla de análisis LL(1)

- **Ejecución exitosa:**  
La tabla de análisis LL(1) se muestra en pantalla.
- **Ejecución errónea:**  
Ver errores de Ver tabla de análisis LL(1).

### 2.1.7. Análisis SLR(1)

#### Previos

- Apertura o edición correcta de la gramática.

#### Resultados tras el análisis SLR(1)

- **Ejecución exitosa:**  
Se muestra en pantalla la gramática ampliada utilizada durante el análisis, la tabla de análisis SLR(1) y el autómata de conjuntos de configuraciones LR(0) en modo gráfico y en modo texto correspondientes a dicha gramática. En la tabla se especifica si la gramática es o no del tipo SLR(1) y las causas de ello.
- **Ejecución errónea:**  
Ver errores del análisis SLR(1).

### 2.1.8. Ver tabla de análisis y el autómata SLR(1)

#### Previos

- Gramática analizada según el tipo SLR(1).

#### Resultados tras ver la tabla y el autómata SLR(1)

- **Ejecución exitosa:**  
La gramática ampliada utilizada durante el análisis, la tabla del análisis SLR(1) y el autómata de conjuntos de configuraciones LR(0) en modo gráfico y en modo texto correspondientes a dicha gramática se muestran en pantalla.
- **Ejecución errónea:**  
Ver errores de Ver tabla de análisis SLR(1).

### 2.1.9. Análisis LR(1)

#### Previos

- Apertura o edición correcta de la gramática.

#### Resultados tras el análisis LR(1)

- **Ejecución exitosa:**  
Se muestra en pantalla la gramática ampliada utilizada durante el análisis, la tabla de análisis LR(1) y el autómata de conjuntos de configuraciones LR(1) en modo gráfico y

en modo texto correspondientes a dicha gramática. En la tabla se especifica si la gramática es o no del tipo LR(1) y las causas de ello.

- **Ejecución errónea:**  
Ver errores del análisis LR(1).

#### **2.1.10. Ver tabla de análisis y el autómata LR(1)**

##### **Previos**

- Gramática analizada según el tipo LR(1).

##### **Resultados tras ver la tabla y el autómata LR(1)**

- **Ejecución exitosa:**  
La gramática ampliada utilizada durante el análisis, la tabla del análisis LR(1) y el autómata de conjuntos de configuraciones LR(1) en modo gráfico y en modo texto correspondientes a dicha gramática se muestran en pantalla.
- **Ejecución errónea:**  
Ver errores de Ver tabla de análisis LR(1).

#### **2.1.11. Análisis LALR(1)**

##### **Previos**

- Apertura o edición correcta de la gramática.

##### **Resultados tras el análisis LALR(1)**

- **Ejecución exitosa:**  
Se muestra en pantalla la gramática ampliada utilizada durante el análisis, la tabla de análisis LALR(1) y el autómata de conjuntos de configuraciones LR(0) en modo gráfico y en modo texto correspondientes a dicha gramática. En la tabla se especifica si la gramática es o no del tipo LALR(1) y las causas de ello.
- **Ejecución errónea:**  
Ver errores del análisis LALR(1).

#### **2.1.12. Ver tabla de análisis y el autómata LALR(1)**

##### **Previos**

- Gramática analizada según el tipo LALR(1).

##### **Resultados tras ver la tabla y el autómata LALR(1)**

- **Ejecución exitosa:**  
La gramática ampliada utilizada durante el análisis, la tabla del análisis LALR(1) y el autómata de conjuntos de configuraciones LR(0) en modo gráfico y en modo texto correspondientes a dicha gramática se muestran en pantalla.
- **Ejecución errónea:**  
Ver errores de Ver tabla de análisis LALR(1).

### **2.1.13. Simular entrada LL(1)**

#### **Previos**

- Gramática analizada según el tipo LL(1).
- Gramática correspondiente al tipo LL(1).
- Aparición de la ventana donde introducimos la frase que va a ser simulada para ver si es reconocida por la gramática.

#### **Resultados tras simular entrada LL(1)**

- **Ejecución exitosa:**  
Se muestra en pantalla las estructuras necesarias en la simulación (pila de símbolos, producciones emitidas y entrada del simulador) y cada vez que se presiona sobre el botón de simular 1 paso se ejecuta un paso de la simulación y se actualizan los valores de las estructuras. Asimismo, a la vez que se ejecuta la simulación se muestra el árbol de sintaxis asociado a la entrada que se está comprobando.  
Se determina si la entrada es aceptada por la gramática según el analizador LL(1) o no.
- **Ejecución errónea:**  
Ver errores de simular entrada LL(1).

### **2.1.14. Simular entrada SLR(1)**

#### **Previos**

- Gramática analizada según el tipo SLR(1).
- Gramática correspondiente al tipo SLR(1).
- Aparición de la ventana donde introducimos la frase que va a ser simulada para ver si es reconocida por la gramática.

#### **Resultados tras simular entrada SLR(1)**

- **Ejecución exitosa:**  
Se muestra en pantalla las estructuras necesarias en la simulación (pila de símbolos, producciones emitidas y entrada del simulador) y cada vez que se presiona sobre el botón de simular 1 paso se ejecuta un paso de la simulación y se actualizan los valores de las estructuras. Asimismo, a la vez que se ejecuta la simulación se muestra el árbol de sintaxis asociado a la entrada que se está comprobando.  
Se determina si la entrada es aceptada por la gramática según el analizador SLR(1) o no.
- **Ejecución errónea:**  
Ver errores de simular entrada SLR(1).

### **2.1.15. Simular entrada LR(1)**

#### **Previos**

- Gramática analizada según el tipo LR(1).
- Gramática correspondiente al tipo LR(1).

- Aparición de la ventana donde introducimos la frase que va a ser simulada para ver si es reconocida por la gramática.

#### Resultados tras simular entrada LR(1)

- **Ejecución exitosa:**  
Se muestra en pantalla las estructuras necesarias en la simulación (pila de símbolos, producciones emitidas y entrada del simulador) y cada vez que se presiona sobre el botón de simular 1 paso se ejecuta un paso de la simulación y se actualizan los valores de las estructuras. Asimismo, a la vez que se ejecuta la simulación se muestra el árbol de sintaxis asociado a la entrada que se está comprobando.  
Se determina si la entrada es aceptada por la gramática según el analizador LR(1) o no.
- **Ejecución errónea:**  
Ver errores de simular entrada LR(1).

### 2.1.16. Simular entrada LALR(1)

#### Previos

- Gramática analizada según el tipo LALR(1).
- Gramática correspondiente al tipo LALR(1).
- Aparición de la ventana donde introducimos la frase que va a ser simulada para ver si es reconocida por la gramática.

#### Resultados tras simular entrada LALR(1)

- **Ejecución exitosa:**  
Se muestra en pantalla las estructuras necesarias en la simulación (pila de símbolos, producciones emitidas y entrada del simulador) y cada vez que se presiona sobre el botón de simular 1 paso se ejecuta un paso de la simulación y se actualizan los valores de las estructuras. Asimismo, a la vez que se ejecuta la simulación se muestra el árbol de sintaxis asociado a la entrada que se está comprobando.  
Se determina si la entrada es aceptada por la gramática según el analizador LALR(1) o no.
- **Ejecución errónea:**  
Ver errores de simular entrada LALR(1).

### 2.1.17. Comprobar entrada

#### Previos

- Gramática analizada, como mínimo, por uno de los cuatro tipos (LL(1), SLR(1), LR(1) y LALR(1)).
- Gramática correspondiente, como mínimo, a uno de los cuatro tipos (LL(1), SLR(1), LR(1) y LALR(1)).
- Aparición de la ventana donde introducimos la frase que va a ser simulada para ver si es reconocida por la gramática, el tipo de analizador que utilizaremos y si queremos obtener el árbol de sintaxis o no.

#### Resultados tras comprobar entrada

- **Ejecución exitosa:**



Se muestra en pantalla si la entrada es reconocida o no por el analizador sintáctico que hemos elegido asociado a la gramática. Asimismo se mostrará una nueva ventana con el árbol de sintaxis obtenido, si así se ha especificado.

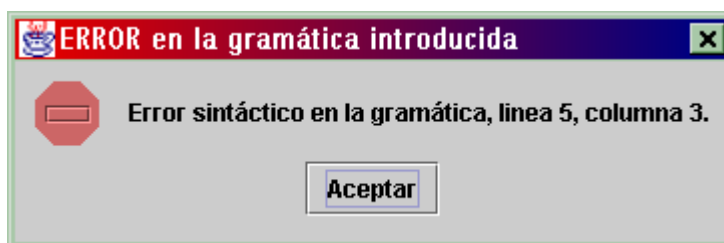
- **Ejecución errónea:**  
Ver errores de comprobar entrada.

## 2.2. Errores y posibles causas

Se muestran ahora una relación de los diferentes tipos de error que se pueden dar al elegir la opción que las precede.

Estos errores están numerados y su especificación consta del nombre del error, cabecera del mensaje que nos aparece por pantalla al producirse un error, la explicación del error en sí, una generalización de las posibles causas por las que se ha producido y el resultado que desencadena.

Al producirse un error aparecerá una ventana indicando el error producido. En dicha ventana habrá un botón de *Aceptar* que desencadenará los resultados producidos por el error. Un ejemplo de esta ventana se muestra en la figura siguiente:



### 2.2.1. Apertura de la gramática

#### ERROR 1: Error sintáctico

**Cabecera del mensaje de error:** 'Error en el fichero de la gramática'.

**Mensaje de error que aparece en pantalla:** 'Error sintáctico en la gramática, línea x, columna y'.

**Causas:** Error sintáctico en el fichero del que se quiere leer la gramática que usará ANAGRA.

**Resultados del error:** Se abre el fichero pero solo se permite su edición. Para poder analizar la gramática y transformarla se deberá de presionar la opción *Aceptar gramática* del menú *Edición* para volver a ver la corrección de la gramática.

#### ERROR 2: Error léxico

**Cabecera del mensaje de error:** 'Error en el fichero de la gramática'.

**Mensaje de error que aparece en pantalla:** 'Error léxico en la gramática, línea x, columna y'.

**Causas:** Error léxico en el fichero del que se quiere leer la gramática que usará ANAGRA.

**Resultados del error:** Se abre el fichero pero solo se permite su edición. Para poder analizar la gramática y transformarla se deberá de presionar la opción *Aceptar gramática* del menú *Edición* para volver a ver la corrección de la gramática.

#### ERROR 3: Token desconocido

**Cabecera del mensaje de error:** 'Error en el fichero de la gramática'.

**Mensaje de error que aparece en pantalla:** 'Token no definido: tk, línea x, columna y'.

**Causas:** Se ha localizado en la gramática que se quiere usar, un símbolo terminal que no ha sido definido como tal en la gramática y es un símbolo no literal.

**Resultados del error:** Se abre el fichero pero solo se permite su edición. Para poder analizar la gramática y transformarla se deberá de presionar la opción *Aceptar gramática* del menú *Edición* para volver a ver la corrección de la gramática.

#### ERROR 4: Fichero no encontrado

**Cabecera del mensaje de error:** 'Error en el fichero de la gramática'.

**Mensaje de error que aparece en pantalla:** 'Fichero no encontrado'.

**Causas:** El fichero que se especificó para su lectura no ha sido encontrado.

**Resultados del error:** Se aborta la operación de apertura del fichero.

### 2.2.2. Edición de la gramática

#### ERROR 1: Error sintáctico

**Cabecera del mensaje de error:** 'Error en la gramática introducida'.

**Mensaje de error que aparece en pantalla:** 'Error sintáctico en la gramática, línea x, columna y'.

**Causas:** Error sintáctico en la gramática que ha sido editada.

**Resultados del error:** Continúa la operación de edición de la gramática hasta que se eliminen los errores o desestime su edición.

#### ERROR 2: Error léxico

**Cabecera del mensaje de error:** 'Error en la gramática introducida'.

**Mensaje de error que aparece en pantalla:** 'Error léxico en la gramática, línea x, columna y'.

**Causas:** Error léxico en la gramática que ha sido editada.

**Resultados del error:** Continúa la operación de edición de la gramática hasta que se eliminen los errores o desestime su edición.

#### ERROR 3: Token desconocido

**Cabecera del mensaje de error:** 'Error en la gramática introducida'.

**Mensaje de error que aparece en pantalla:** 'Token no definido: tk, línea x, columna y'.

**Causas:** Se ha localizado en la gramática que ha sido editada, un símbolo terminal que no ha sido definido como tal en la gramática y es un símbolo no literal.

**Resultados del error:** Continúa la operación de edición de la gramática hasta que se eliminen los errores o desestime su edición.

### 2.2.3. Cálculo del conjunto PRI de una forma de frase

#### ERROR 1: Forma de frase incorrecta

**Cabecera del mensaje de error:** 'Error en la forma de frase'.

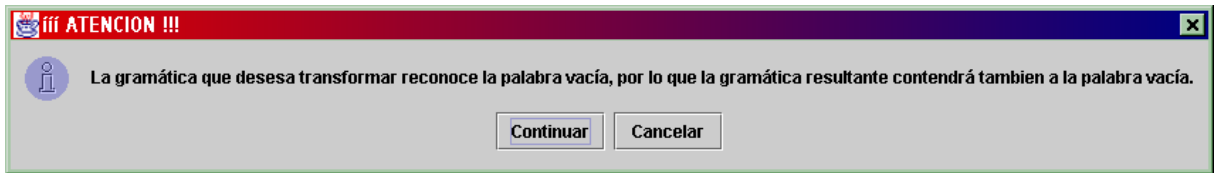
**Mensaje de error que aparece en pantalla:** 'Símbolo x no perteneciente a la gramática'.

**Causas:** Algún símbolo de la forma de frase introducida no pertenece a la gramática.

**Resultados del error:** Continúa activa la ventana de obtención del conjunto PRI de una forma de frase a la espera de que la forma de frase introducida sea correcta.

### 2.2.4. Transformaciones de la gramática

Al realizar la transformación de eliminación de lo anulables puede aparecer el siguiente mensaje de aviso:



**AVISO1: La gramática reconoce a la palabra vacía en la eliminación de anulables**

**Cabecera del mensaje de error:** '¡¡¡ ATENCION !!!'.

**Mensaje de error que aparece en pantalla:** 'La gramática que desea transformar reconoce la palabra vacía, por lo que la gramática resultante contendrá también a la palabra vacía'.

**Causas:** La gramática que se desea transformar reconoce a la palabra vacía.

**Resultados del aviso:** Si se elige la opción de continuar la transformación, la gramática resultante reconocerá también a la palabra vacía, siendo el símbolo inicial el único no terminal anulable de la gramática. En caso contrario de se anulará la transformación y se volverá a la pantalla inicial.

**2.2.5. Simular entrada LL(1)**

**ERROR 1: Entrada no reconocida por la gramática**

**Cabecera del mensaje de error:** 'Error en la entrada del simulador'.

**Mensaje de error que aparece en pantalla:** 'La entrada NO es reconocida por la gramática'.

**Causas:** La forma de frase que introducimos no es reconocida por la gramática.

**Resultados del error:** Se mantiene visible la ventana del simulador para que se puedan observar las distintas estructuras con el fin de razonar el error producido. La única posibilidad que se puede realizar en este momento es cerrar la ventana del simulador.

**2.2.6. Simular entrada SLR(1)**

**ERROR 1: Entrada no reconocida por la gramática**

**Cabecera del mensaje de error:** 'Error en la entrada del simulador'.

**Mensaje de error que aparece en pantalla:** 'La entrada NO es reconocida por la gramática'.

**Causas:** La forma de frase que introducimos no es reconocida por la gramática.

**Resultados del error:** Se mantiene visible la ventana del simulador para que se puedan observar las distintas estructuras con el fin de razonar el error producido. La única posibilidad que se puede realizar en este momento es cerrar la ventana del simulador.

**2.2.7. Simular entrada LR(1)**

**ERROR 1: Entrada no reconocida por la gramática**

**Cabecera del mensaje de error:** 'Error en la entrada del simulador'.

**Mensaje de error que aparece en pantalla:** 'La entrada NO es reconocida por la gramática'.

**Causas:** La forma de frase que introducimos no es reconocida por la gramática.

**Resultados del error:** Se mantiene visible la ventana del simulador para que se puedan observar las distintas estructuras con el fin de razonar el error producido. La única posibilidad que se puede realizar en este momento es cerrar la ventana del simulador.

### **2.2.8. Simular entrada LALR(1)**

#### **ERROR 1: Entrada no reconocida por la gramática**

**Cabecera del mensaje de error:** 'Error en la entrada del simulador'.

**Mensaje de error que aparece en pantalla:** 'La entrada NO es reconocida por la gramática'.

**Causas:** La forma de frase que introducimos no es reconocida por la gramática.

**Resultados del error:** Se mantiene visible la ventana del simulador para que se puedan observar las distintas estructuras con el fin de razonar el error producido. La única posibilidad que se puede realizar en este momento es cerrar la ventana del simulador.

### **2.2.9. Comprobar entrada**

#### **ERROR 1: Entrada no reconocida por la gramática**

**Cabecera del mensaje de error:** 'Error en la entrada del simulador'.

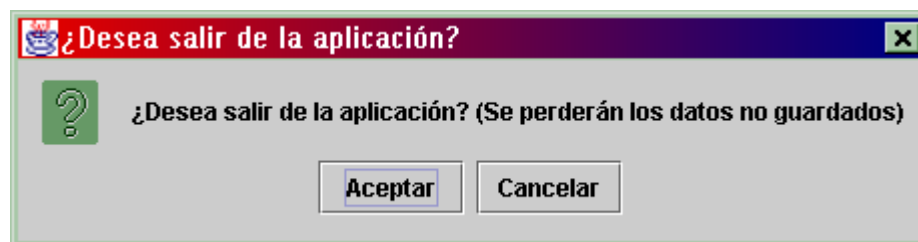
**Mensaje de error que aparece en pantalla:** 'La entrada NO es reconocida por la gramática'.

**Causas:** La forma de frase que introducimos no es reconocida por la gramática.

**Resultados del error:** La única posibilidad que se puede realizar en este momento es cerrar la ventana donde se muestra la forma de frase a simular y el tipo de analizador.

### **2.2.10. Salir de la aplicación**

Cuando salimos de la aplicación con la opción *Salir* del menú *Gramática*, aparece un mensaje en el que se advierte que todas las ventanas de la aplicación serán cerradas perdiéndose los datos que no hayan sido guardados en disco.



### **3. Tutorial del sistema**

Una vez que la aplicación está en funcionamiento el usuario se encuentra ante una pantalla que a partir de ahora denominaremos *Pantalla Principal* (ver Figura 1).

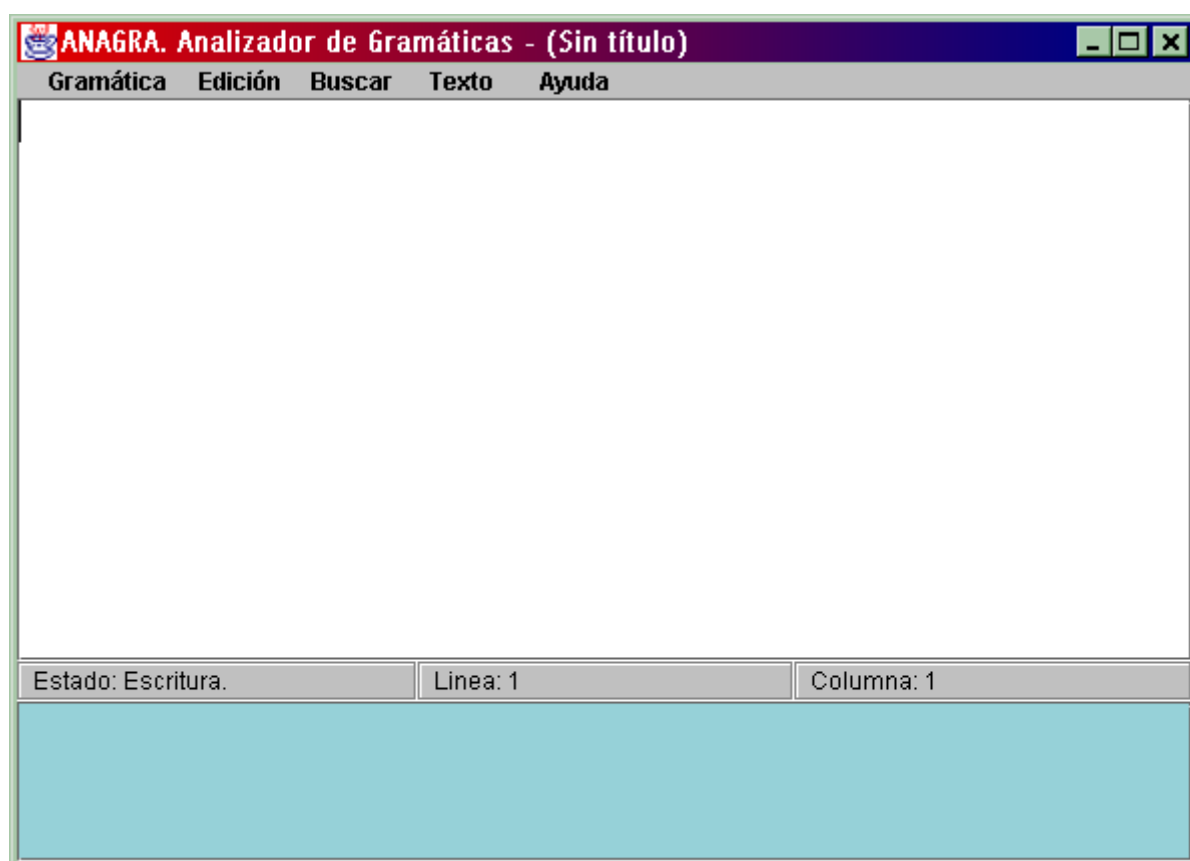


Figura 1: Pantalla Principal.

Esta pantalla consta de una serie de menús que ofrecen al usuario todas las posibilidades de la aplicación, un cuadro de texto donde se muestra la gramática con la que estamos trabajando, una barra de estado donde se muestra el estado del editor (escritura, lectura) y la línea y columna donde se encuentra el cursor, y un cuadro de texto en la parte inferior donde se va informando de las operaciones que se realizan.

Los menús desplegables que aparecen en la *Pantalla Principal* se detallan en los puntos siguientes.

### 3.1. Menú Gramática

Lo primero que tenemos que hacer para trabajar con ANAGRA es disponer de una gramática sobre la que poder operar. Para ello tenemos dos posibilidades: la podemos obtener desde un fichero o la podemos escribir nosotros mismos.

#### 3.1.1. Nueva (CTRL+N)

Esta opción crea una ventana nueva independiente de la actual donde se puede trabajar con otra gramática distinta.

#### 3.1.2. Abrir (CTRL+O)

Esta opción permite obtener la gramática con la que vamos a trabajar de un fichero que seleccionaremos de un cuadro de diálogo de apertura de fichero. El formato de dicho fichero aparece en el punto 3.10. **Formatos de entrada** del presente documento. En la figura 2 aparece un cuadro de diálogo de apertura de fichero:

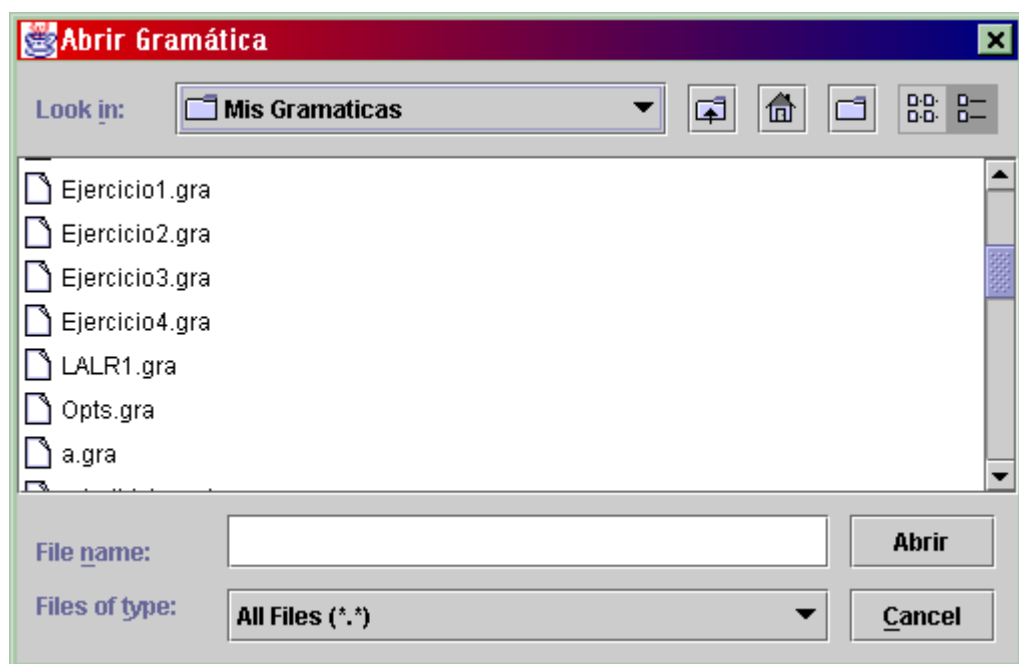


Figura 2: Diálogo de apertura de fichero

Una vez que hayamos seleccionado la gramática y esta no sea errónea, se mostrará en la Pantalla Principal.

#### 3.1.3. Editar (CTRL+E)

La segunda posibilidad que tenemos es escribir la gramática nosotros mismos. Para ello utilizaremos esta opción del menú. Al arrancar la aplicación ya estamos en modo edición. Utilizaremos esta opción para modificar una gramática abierta anteriormente o editada

anteriormente y que ya hayan sido comprobadas. Cuando una gramática se comprueba que es correcta pasa a modo lectura para poder actuar con ella.

#### **3.1.4. Cerrar (CTRL+W)**

Con esta opción se cierra la ventana actual perdiéndose toda aquella información que no haya sido guardada.

#### **3.1.5. Guardar (CTRL+G)**

Con esta opción se guarda la gramática actual de trabajo en disco. Si la gramática ya había sido guardada previamente o había sido leída de disco, se guardará automáticamente. En caso contrario de comportará como la opción *Guardar como* explicada en el apartado siguiente.

#### **3.1.6. Guardar como**

Esta opción permite lo mismo que la anterior pero dando la opción de que el fichero donde se guarde la gramática sea introducido. En la figura 3 aparece un diálogo de ejemplo:

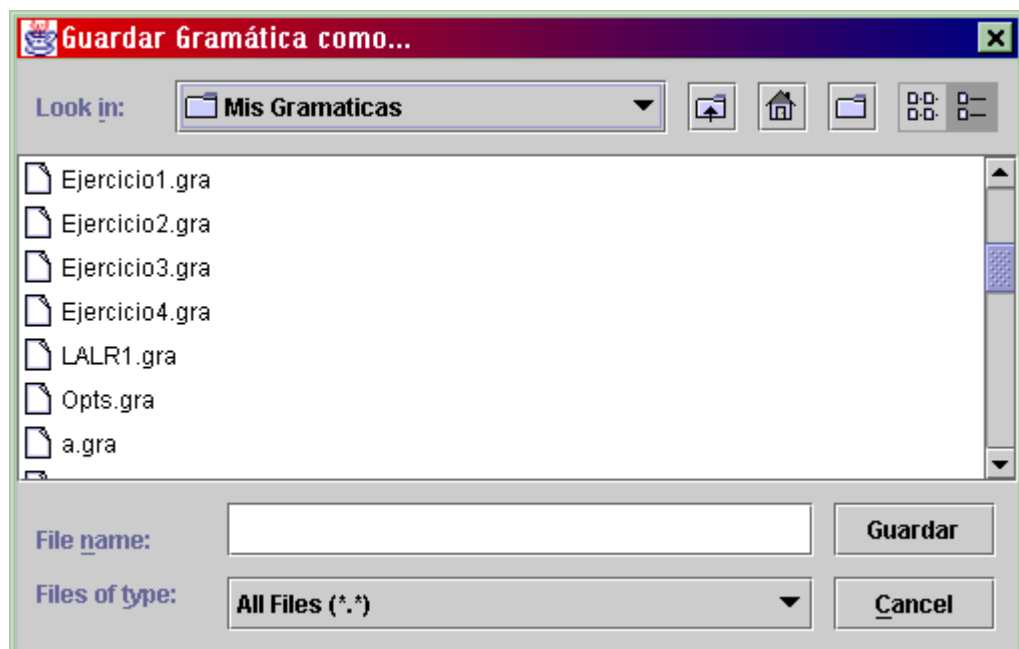


Figura 3: Diálogo guardar como...

#### **3.1.7. Salir (CTRL + Q)**

Provoca el cierre de todas las ventanas abiertas de la aplicación y la pérdida de toda aquella información que no haya sido guardada. Aparece un diálogo recordándonos esto último y dándonos la posibilidad de continuar la operación o cancelarla. En la figura 4 aparece este diálogo:



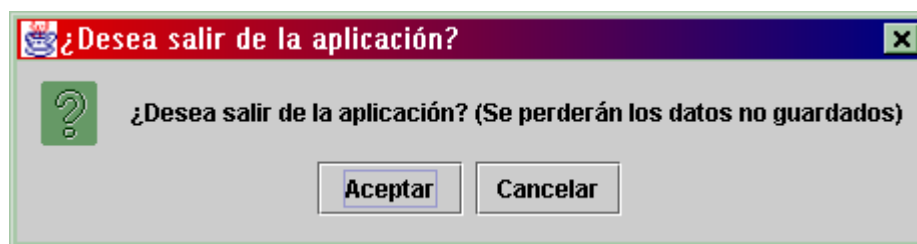


Figura 4: Diálogo de salida de la aplicación

## 3.2. Menú Edición

Este menú solo aparece presente cuando se ha activado el modo edición presionado la opción *Editar* del menú *Gramática*.

### 3.2.1. Cortar (CTRL + X)

Borra el trozo de texto seleccionado del área de edición de la gramática y lo copia en el portapapeles del sistema.

### 3.2.2. Copiar (CTRL + C)

Copia el trozo de texto seleccionado del área de edición de la gramática al portapapeles del sistema.

### 3.2.3. Pegar (CTRL + V)

Pega el contenido del portapapeles en el área de edición de la gramática.

### 3.2.4. Borrar

Borra el trozo de texto seleccionado.

### 3.2.5. Seleccionar todo (CTRL+A)

Selecciona todo el texto introducido en el área de edición de gramáticas.

### 3.2.7. Aceptar Gramática

Se ha de presionar esta opción cuando se haya terminado de editar la gramática y se quiera poder utilizar las opciones de análisis y transformación sobre ella. Una vez presionado se verificará la corrección de la gramática editada. Si es correcta finalizará el modo edición. En caso contrario se mostrará un mensaje de error, permaneciendo en el modo edición.

### 3.2.8. Desestimar Gramática

Esta opción sirve para olvidar los cambios hechos durante la edición de la gramática y volver a la gramática antes de ser editada. Si no había ninguna gramática anterior no hace nada.

## 3.3. Menú Buscar

### 3.3.1. Buscar (CTRL + B)

Esta opción muestra el cuadro de diálogo de la figura 5. Sirve para realizar la búsqueda de determinado texto siguiendo diferentes opciones.

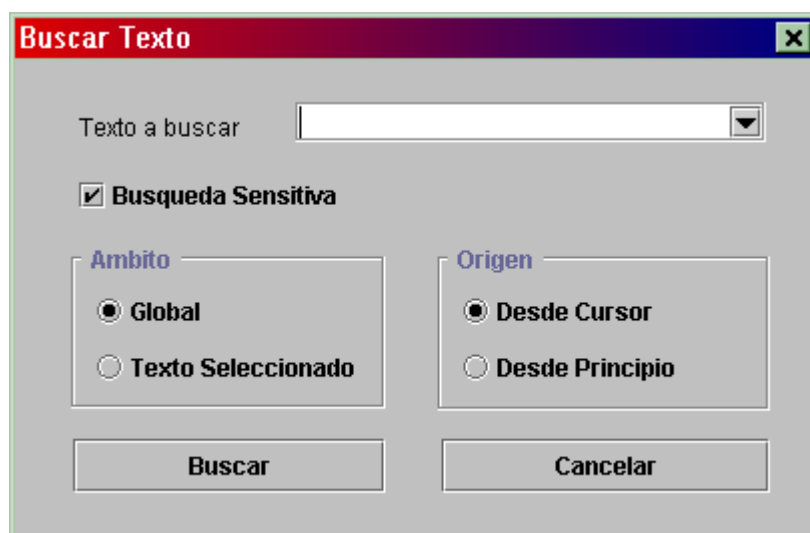


Figura 5: Diálogo Buscar

El cuadro de diálogo guarda las diez últimas búsquedas realizadas.

### **3.3.2. Reemplazar (CTRL + R)**

Esta opción muestra el cuadro de diálogo de la figura 6. Sirve para realizar el reemplazamiento de determinado texto siguiendo diferentes opciones.

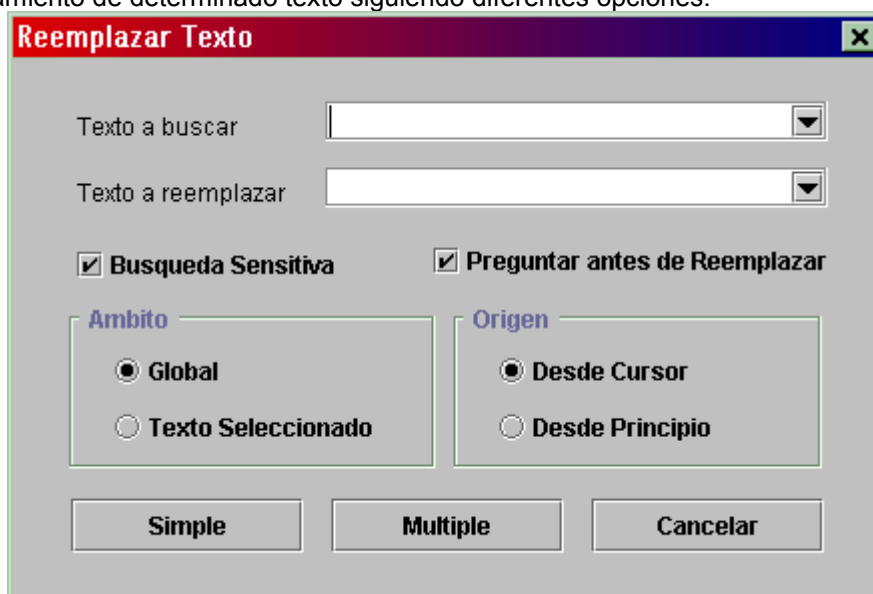


Figura 6: Diálogo Reemplazar

El cuadro de diálogo guarda los diez últimos reemplazamientos.  
Esta opción solamente está activa en modo edición.

### **3.3.3. Buscar de nuevo (SHIFT + B)**

Esta opción sirve para buscar de nuevo la última palabra buscada mediante la opción de búsqueda a partir de la posición actual del cursor.

### **3.3.4. Ir a línea**

Esta opción permite al usuario ir a una determinada línea de la gramática. La figura 7 muestra el diálogo en cuestión.

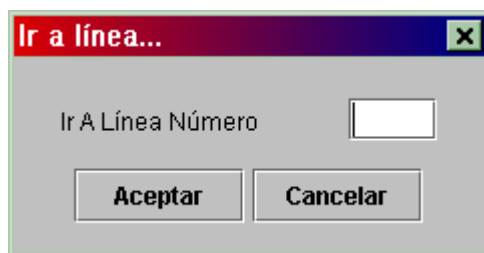


Figura 7: Diálogo Ir a línea...

## **3.4. Menú Texto**

En este menú aparecen opciones de cambio de tipo de letra, de color, etc..

### **3.4.1. Fuente**

Permite cambiar el tipo de letra con el que estamos editando o mostrando la gramática. En la figura 8 aparece el diálogo.

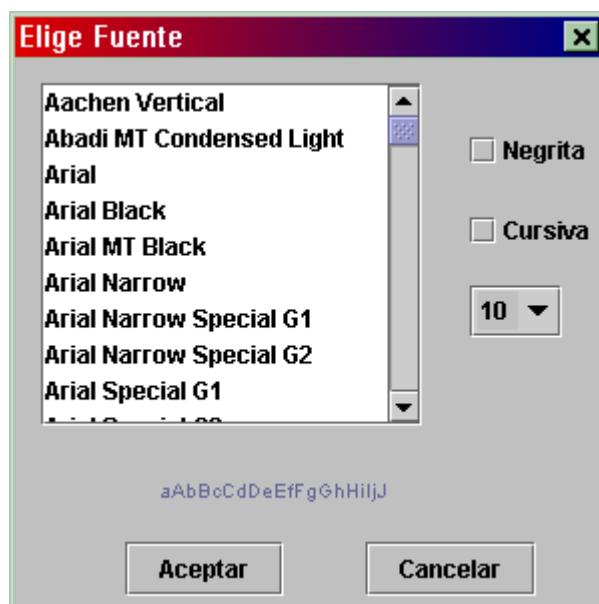


Figura 8: Diálogo Fuente

### **3.4.2. Color**

Permite cambiar el color de letra con el que estamos editando o mostrando la gramática. En la figura 9 aparece el diálogo.

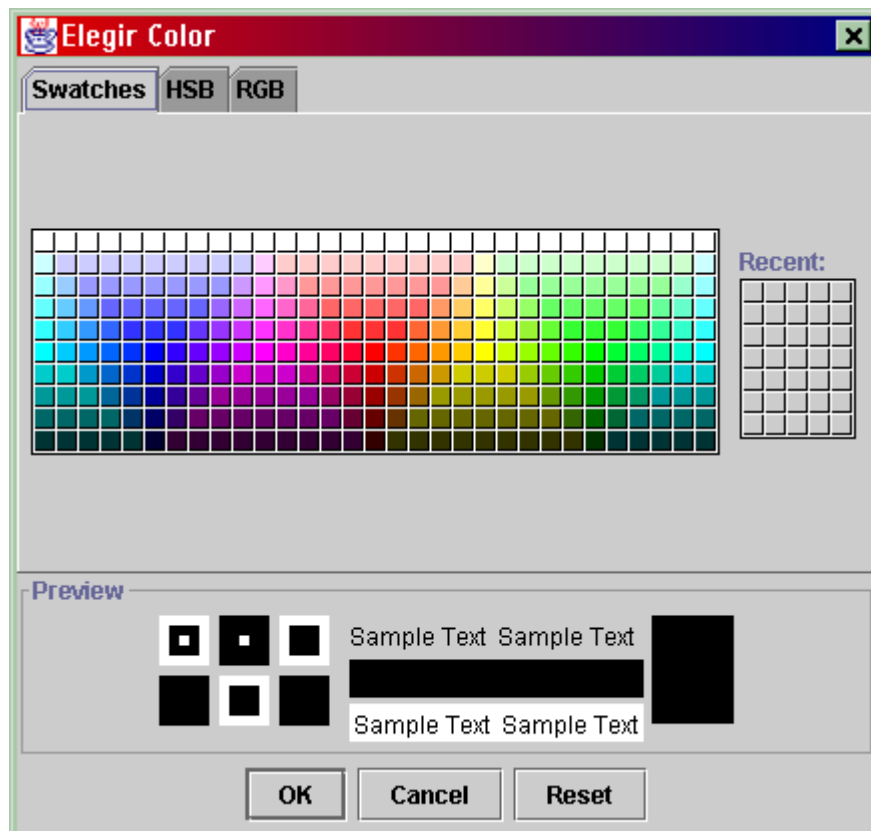


Figura 9: Diálogo Color

### **3.4.3. Tabulador**

Permite cambiar el número de espacios que formarán un tabulador por defecto.

### **3.4.4. Extendido**

Permite cambiar el formato en el que aparece la gramática editada o abierta al formato extendido. En este formato las producciones aparecen de la siguiente manera.

*Parte Izquierda :*

*Parte Derecha1*  
 | *Parte Derecha2*  
 | .....  
 | *Parte Derecha n*

;

### **3.4.5. Compacto**

Permite cambiar el formato en el que aparece la gramática editada o abierta al formato compacto. En este formato las producciones aparecen de la siguiente manera.

*Parte Izquierda : Parte Derecha1 | Parte Derecha2 | ..... | Parte Derecha n;*

### **3.4.6. Guardar Preferencias**

Permite guardar las opciones de tipo de letra, color de letra y espacios de tabulador elegidos, para que la próxima vez que se cargue la aplicación sean tomados como valores por defecto.

### 3.5. Menú Herramientas

#### 3.5.1. Calcular Conjunto PRIMERO

Muestra una nueva ventana donde se muestra el conjunto de PRIMERO de los símbolos de una gramática previamente abierta o editada.

#### 3.5.2. Calcular Conjunto SIGUIENTE

Muestra una nueva ventana donde se muestra el conjunto de SIGUIENTE de los símbolos de una gramática previamente abierta o editada.

#### 3.5.3. Calcular PRIMERO de una forma de frase

Esta opción permite calcular el conjunto PRI de una forma de frase formada por símbolos pertenecientes a la gramática. Aparecerá una ventana donde en el campo superior escribiremos la forma de frase símbolo a símbolo separados por uno o más espacios o tabuladores (para consultar el formato de entrada de la forma de frase a introducir nos referimos al punto **3.10. Formatos de entrada**). Una vez que hayamos escrito la forma de frase, pulsaremos el botón *Calcular*, y el conjunto PRI de símbolos aparecerá en el campo inferior de la ventana. El símbolo *EPS* corresponde a *Epsilon*. En la figura 10 parece la ventana:

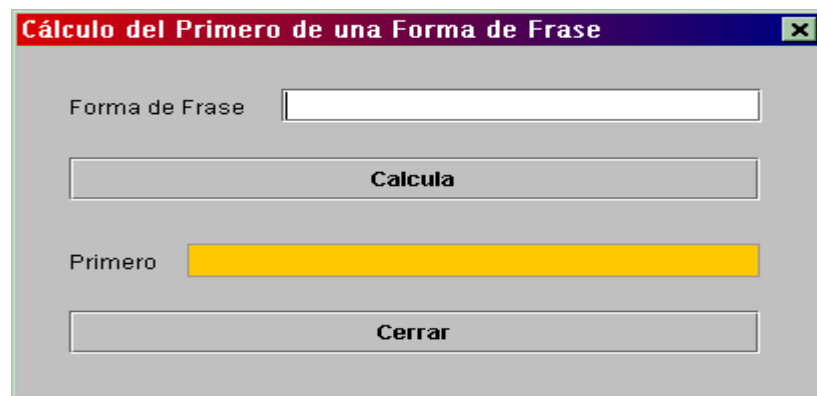


Figura 10: Ventana de cálculo del conjunto PRI de una forma de frase

### 3.6. Menú Transformaciones

En este menú aparece la posibilidad de aplicar distintas transformaciones a la gramática actual para convertirla en otra equivalente con objeto de que se pueda adaptar a uno u otro tipo de gramática. Las posibles transformaciones se enumeran a continuación:

- Eliminación de los símbolos que no son capaces de derivar ningún otro símbolo (Eliminación de no derivables).
- Factorización a izquierda de todas aquellas producciones que pueden provocar **backtracking** en el analizador (Factorización a izquierda).
- Eliminación de todos aquellos símbolos no terminales de la gramática que derivan en sí mismos en uno o más pasos (Eliminación de ciclos).
- Eliminación de todos aquellos símbolos de la gramática que no pueden ser accedidos desde el símbolo inicial (Eliminación de no accesibles).
- Eliminación de todas aquellas producciones cuya parte derecha es lambda (Eliminación de anulables).
- Eliminación de la recursividad a izquierda de las producciones, que provoca que una gramática no sea del tipo LL(1) (Eliminación de la recursividad a izquierda).

Para una mayor información acerca de estas transformaciones se ruega conferir los apuntes de la asignatura Compiladores I o el Documento de diseño de objetos. En estos documentos se explica claramente cuál es el objetivo de estas transformaciones y los algoritmos empleados en ellas.

Una vez aplicada la transformación, la gramática resultante aparece en una ventana nueva para que puedan compararse las diferencias entre la gramática original y la transformada. Del mismo modo, si la transformación provoca una eliminación de símbolos gramaticales, estos símbolos eliminados aparecen en la parte inferior de la nueva ventana para información del usuario. La figura 11 representa la nueva ventana donde aparece una gramática a la que se le han eliminado los símbolos que no son accesibles desde el símbolo inicial:

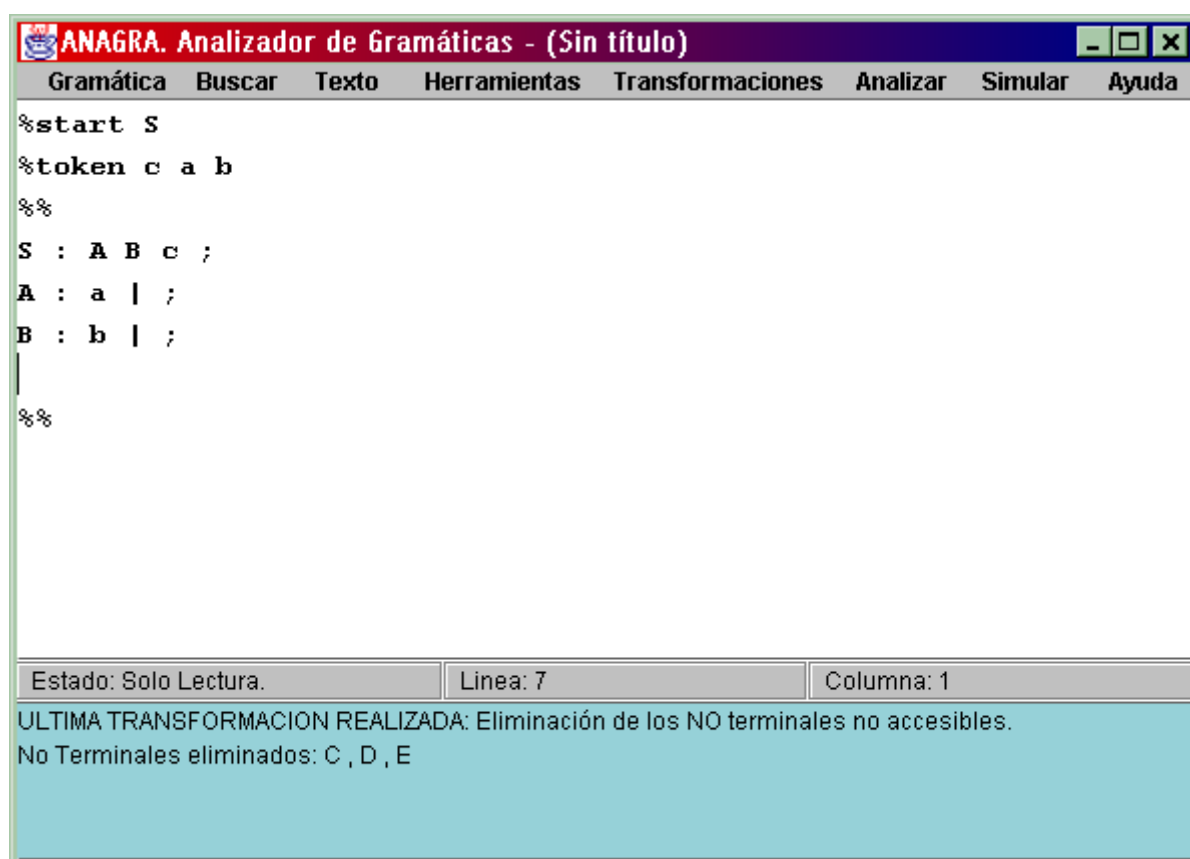


Figura 11: Ventana donde se muestra una gramática transformada

### 3.7. Menú Analizar

Con este menú se puede analizar la gramática actual para determinar el tipo al que pertenece. Los resultados que se obtiene según el tipo de análisis que se aplica se muestran a continuación.

#### 3.7.1. Analizar Gramática LL(1)

Esta opción ejecuta el análisis LL(1) de la gramática, mostrando la tabla obtenida de dicho análisis en una ventana independiente. En la parte inferior de la tabla se indica si la gramática pertenece o no al tipo LL(1), además de unas estadísticas donde se muestran el número de terminales, el número de no terminales y el número de celdas en conflicto. Las casillas de la tabla que aparecen con el fondo rojo son aquellas en las que existe un conflicto que impide que la gramática sea del tipo LL(1). En la figura 12 aparece una de estas tablas generadas a partir del análisis LL(1) de la gramática.

The screenshot shows a window titled 'Tabla LL(1)' with a table of LR(1) items. The columns are labeled with terminals: u, v, w, x, y, z, and \$. The rows are labeled with non-terminals: S, B, D, E, and F. The cell at row B, column w is highlighted in red and contains two productions:  $B \rightarrow Bv$  and  $B \rightarrow w$ , indicating a conflict. Below the table, a message box states: 'ERROR: La gramática no es del tipo LL(1). La entrada de la tabla con coordenadas [B,w] contiene más de una producción.' A 'Cerrar' button is at the bottom.

	u	v	w	x	y	z	\$
S	$S \rightarrow u B D z$						
B			$B \rightarrow Bv$ $B \rightarrow w$				
D				$D \rightarrow E F$	$D \rightarrow E F$	$D \rightarrow E F$	
E				$E \rightarrow EPS$	$E \rightarrow y$	$E \rightarrow EPS$	
F				$F \rightarrow x$		$F \rightarrow EPS$	

ERROR: La gramática no es del tipo LL(1). La entrada de la tabla con coordenadas [B,w] contiene más de una producción.

Cerrar

Figura 12: Tabla obtenida a partir del análisis LL(1)

#### 3.7.2. Ver tabla análisis LL(1)

Esta opción está solamente activa si hemos ejecutado el análisis LL(1) anteriormente, mostrando la tabla que se genera a partir de dicho análisis anterior.

#### 3.7.3. Analizar Gramática SLR(1)

Esta opción ejecuta el análisis SLR(1) de la gramática, mostrando, en ventanas independientes, la tabla obtenida de dicho análisis, el autómata de conjuntos de configuraciones LR(0) a partir del cual se ha obtenido la tabla, la gramática ampliada a partir de la cual se ha obtenido el autómata y una versión del autómata en modo texto. En la parte inferior de la tabla se indica si la gramática pertenece o no al tipo SLR(1), además de unas estadísticas donde se muestran el número de terminales, el número de no terminales, el número de celdas en conflicto en la tabla acción y en la tabla ir\_A, y el número de nodos. Las casillas de la tabla que aparecen con el fondo rojo son aquellas en las que existe un conflicto que impide que la gramática sea del tipo SLR(1). En la figura 13 aparece una de estas tablas generadas a partir del análisis SLR(1) de la gramática.

En otra de las ventanas aparece el autómata de conjuntos de configuraciones LR(0) a partir del cual se genera la tabla. Cuando aparece el autómata los nodos ya están distribuidos por la ventana evitando cruces entre ejes en la medida de lo posible. En estos nodos solo aparece el número del conjunto de configuraciones. Para visualizar la información que se almacena en



cada uno de estos nodos basta con hacer doble click con el ratón sobre el nodo que queremos obtener información. Aparecerá una ventana independiente conteniendo esta información.

Tabla SLR(1)				
Tabla Acción				
	a	b	c	\$
0	d 3	r 4		
1				Aceptar
2		d 5		
3		r 3		
4			r 6 d 7	r 6 r 8
5			r 5	r 5
6				r 2
7				r 7

Tabla Ir A				
	S	A	B	C
0	1	2		
1				
2			4	
3				
4				6
5				
6				
7				

**ERROR:** La gramática no es del tipo SLR(1).Existen entradas en la tabla con conflictos.

**ESTADISTICAS:**

Cerrar

Figura 13: Tabla obtenida a partir del análisis SLR(1)

En la figura 14 se muestra el autómata y en la figura 15 se muestra la ventana que aparece cuando solicito información acerca del conjunto de configuraciones 0.

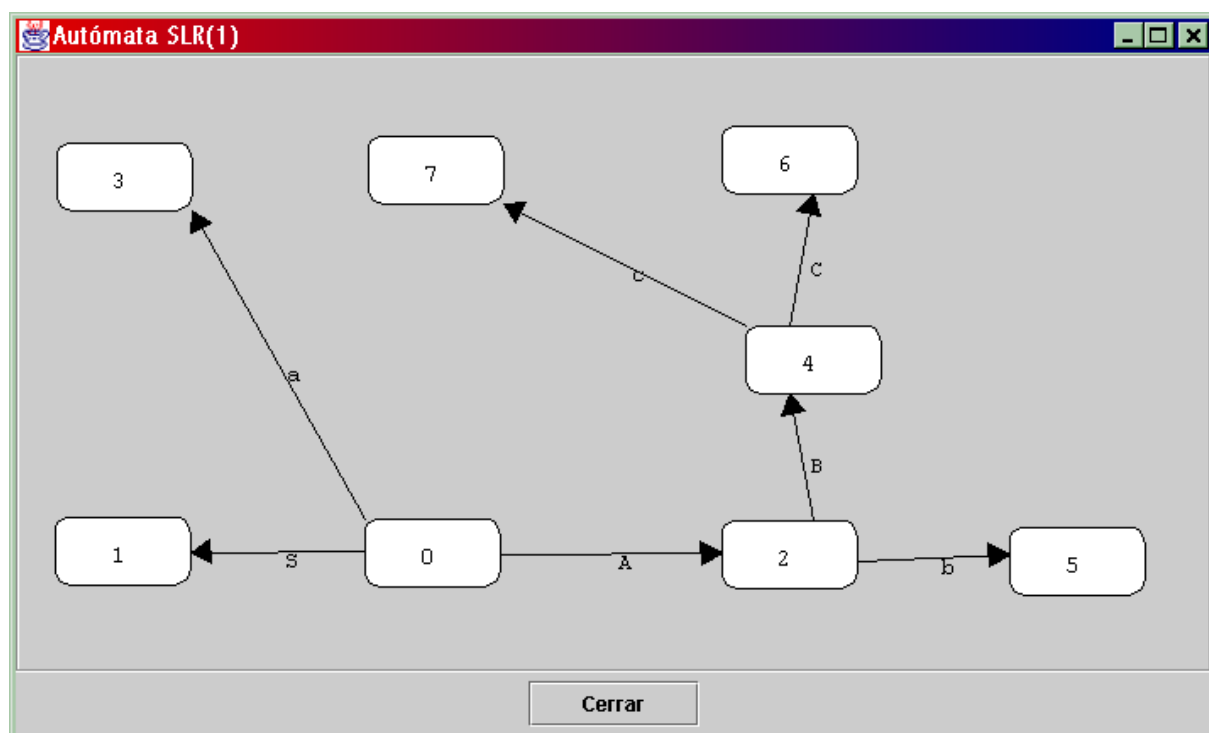


Figura 14: Autómata a partir del cual se genera la tabla SLR(1)

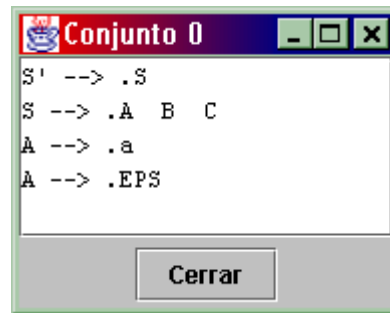


Figura 15: Información acerca del conjunto de configuraciones 0

Además aparecen dos ventanas más conteniendo el autómata en modo texto y la gramática ampliada a partir de la cual se ha generado el autómata. En las figuras 16 y 17 se muestran dichas ventanas.

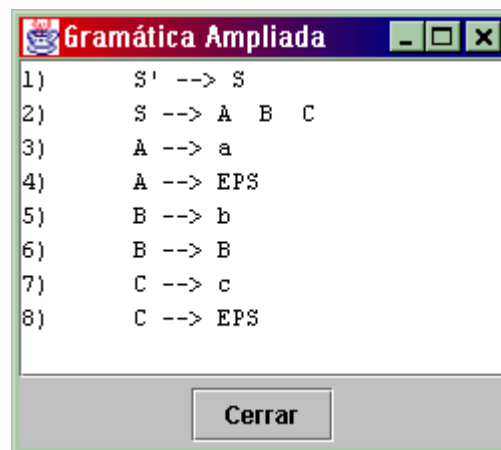


Figura 16: Gramática ampliada

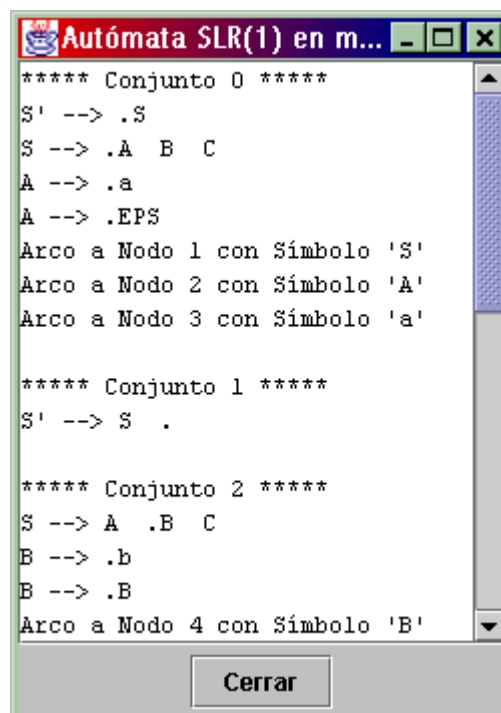


Figura 17: Autómata en modo texto

### 3.7.4. Ver tabla análisis SLR(1)

Esta opción está solamente activa si hemos ejecutado el análisis SLR(1) anteriormente, y muestra las estructuras descritas en el punto anterior.

### 3.7.5. Analizar Gramática LR(1)

Esta opción ejecuta el análisis LR(1) de la gramática, mostrando, en ventanas independientes, la tabla obtenida de dicho análisis, el autómata de conjuntos de configuraciones LR(1) a partir del cual se ha obtenido la tabla, la gramática ampliada a partir de la cual se ha obtenido el autómata y una versión del autómata en modo texto. En la parte inferior de la tabla se indica si la gramática pertenece o no al tipo LR(1), además de unas estadísticas donde se muestran el número de terminales, el número de no terminales, el número de celdas en conflicto en la tabla acción y en la tabla ir\_A, y el número de nodos. Las casillas de la tabla que aparecen con el fondo rojo son aquellas en las que existe un conflicto que impide que la gramática sea del tipo LR(1). En la figura 18 aparece una de estas tablas generadas a partir del análisis LR(1) de la gramática.

Tabla LR(1)												
Tabla Acción						Tabla Ir A						
	NUM	N	,	\n	\$	ALS	AL	NS				
0	d 3					0	1	2				
1	d 3				Aceptar	1		4				
2	r 3				r 3	2						
3		d 6				3			5			
4	r 2				r 2	4						
5		d 8	d 7			5						
6		r 6	r 6			6						
7		d 10				7			9			
8		r 5	r 5			8						
9		d 12		d 11		9						
10		r 6		r 6		10						
11	r 4				r 4	11						
12		r 5		r 5		12						

CORRECTO: La gramática es del tipo LR(1). No existen conflictos en las entradas de la tabla.

ESTADÍSTICAS:

Cerrar

Figura 18: Tabla obtenida a partir del análisis LR(1)

En otra de las ventanas aparece el autómata de conjuntos de configuraciones LR(1) a partir del cual se genera la tabla. Cuando aparece el autómata los nodos ya están distribuidos por la ventana evitando cruces entre ejes en la medida de lo posible. En estos nodos solo aparece el número del conjunto de configuraciones. Para visualizar la información que se almacena en cada uno de estos nodos basta con hacer doble click con el ratón sobre el nodo que queremos obtener información. Aparecerá una ventana independiente conteniendo esta información.

En la figura 19 se muestra el autómata y en la figura 20 se muestra la ventana que aparece cuando solicito información acerca del conjunto de configuraciones 0.

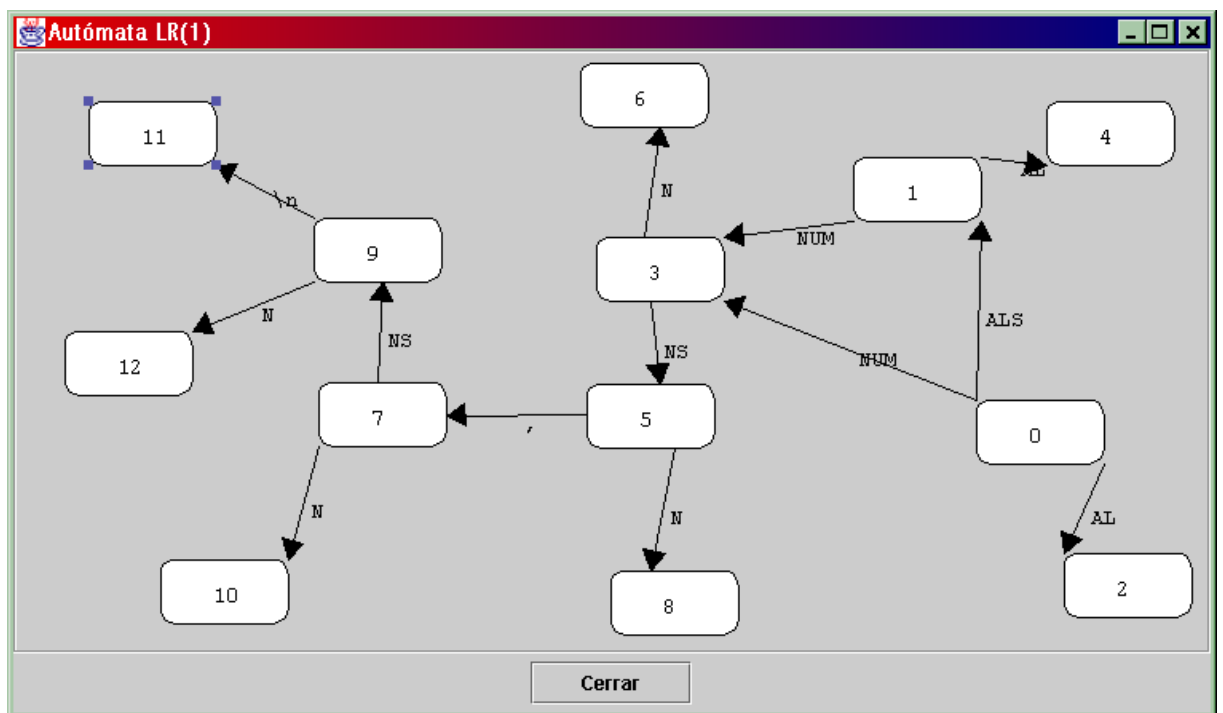


Figura 19: Autómata a partir del cual se genera la tabla LR(1)

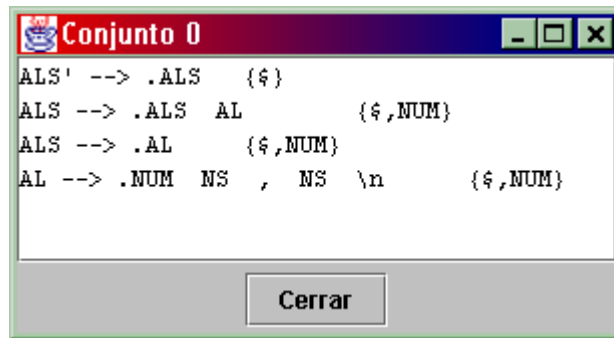


Figura 20: Información acerca del conjunto de configuraciones 0

Además aparecen dos ventanas más conteniendo el autómata en modo texto y la gramática ampliada a partir de la cual se ha generado el autómata. En las figuras 21 y 22 se muestran dichas ventanas.

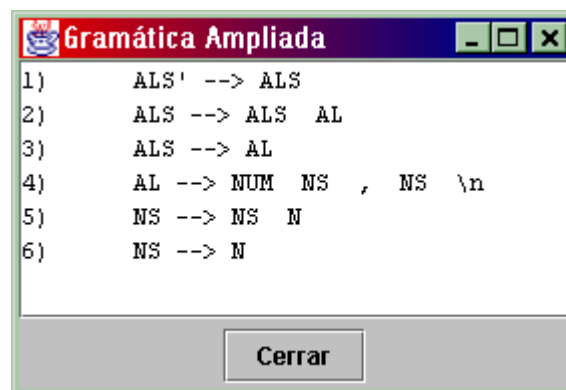


Figura 21: Gramática ampliada

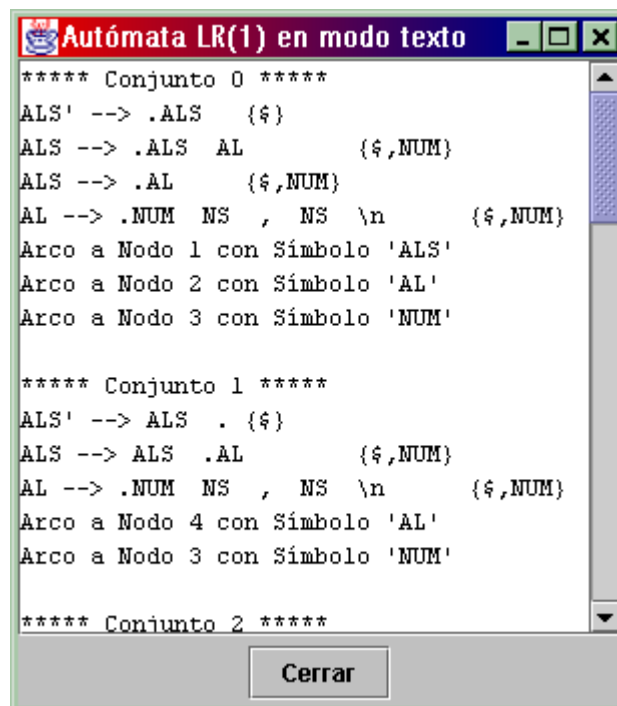


Figura 22: Autómata en modo texto

### 3.7.6. Ver tabla análisis LR(1)

Esta opción está solamente activa si hemos ejecutado el análisis SLR(1) anteriormente, y muestra las estructuras descritas en el punto anterior.

### 3.7.7. Analizar Gramática LALR(1)

Esta opción ejecuta el análisis LALR(1) de la gramática, mostrando, en ventanas independientes, la tabla obtenida de dicho análisis, el autómata de conjuntos de configuraciones LR(0) a partir del cual se ha obtenido la tabla, la gramática ampliada a partir de la cual se ha obtenido el autómata y una versión del autómata en modo texto. En la parte inferior de la tabla se indica si la gramática pertenece o no al tipo LALR(1), además de unas estadísticas donde se muestran el número de terminales, el número de no terminales, el número de celdas en conflicto en la tabla acción y en la tabla ir\_A, y el número de nodos. Las casillas de la tabla que aparecen con el fondo rojo son aquellas en las que existe un conflicto que impide que la gramática sea del tipo LALR(1). En la figura 23 aparece una de estas tablas generadas a partir del análisis LALR(1) de la gramática.

En otra de las ventanas aparece el autómata de conjuntos de configuraciones LR(0) a partir del cual se genera la tabla. Cuando aparece el autómata los nodos ya están distribuidos por la ventana evitando cruces entre ejes en la medida de lo posible. En estos nodos solo aparece el número del conjunto de configuraciones. Para visualizar la información que se almacena en cada uno de estos nodos basta con hacer doble click con el ratón sobre el nodo que queremos obtener información. Aparecerá una ventana independiente conteniendo esta información.

En la figura 24 se muestra el autómata y en la figura 25 se muestra la ventana que aparece cuando solicito información acerca del conjunto de configuraciones 0.

Además aparecen dos ventanas más conteniendo el autómata en modo texto y la gramática ampliada a partir de la cual se ha generado el autómata. En las figuras 26 y 27 se muestran dichas ventanas.

**Tabla LALR(1)**

Tabla Acción						Tabla Ir A	
	a	(	)	,	\$	S	L
0	d 3	d 2				1	
1					Aceptar		
2	d 3	d 2				5	4
3			r 3	r 3	r 3		
4			d 6	d 7			
5			r 5	r 5			
6			r 2	r 2	r 2		
7	d 3	d 2				8	
8			r 4	r 4			

**CORRECTO:** La gramática es del tipo LALR(1). No existen conflictos en las entradas de la tabla.

**ESTADÍSTICAS:**

Cerrar

Figura 23: Tabla obtenida a partir del análisis LALR(1)

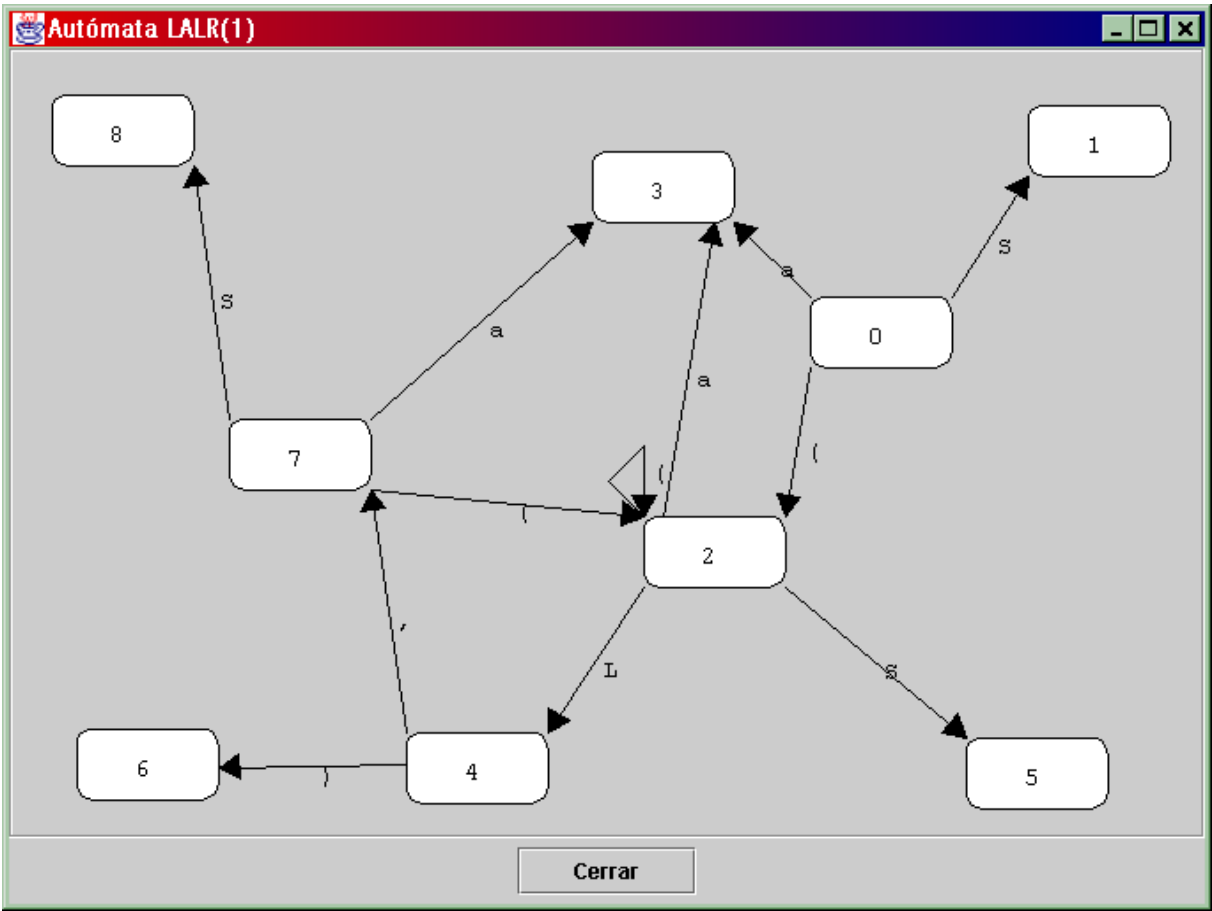




Figura 24: Autómata a partir del cual se genera la tabla LALR(1)

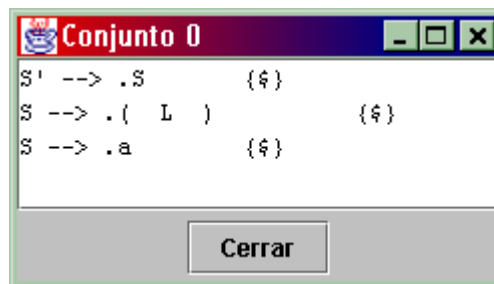


Figura 25: Información acerca del conjunto de configuraciones 0

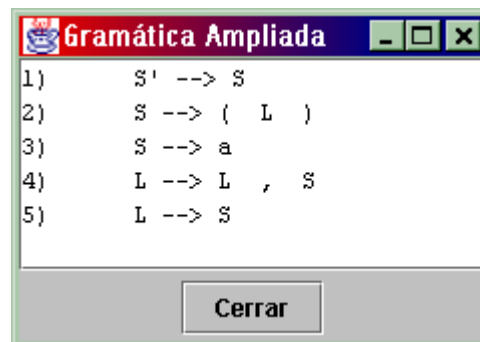


Figura 26: Gramática ampliada

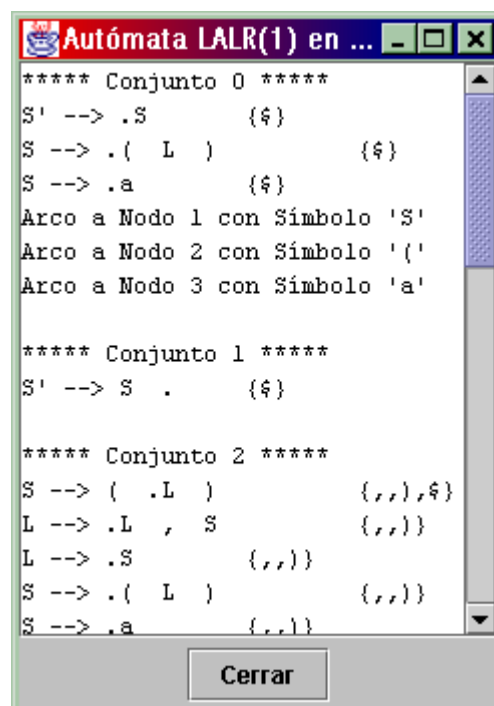


Figura 27: Autómata en modo texto

### 3.7.8. Ver tabla análisis LALR(1)

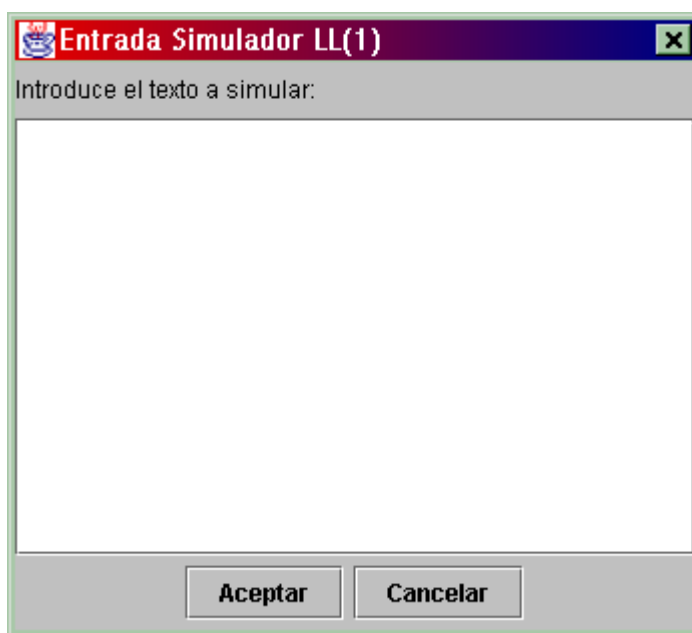
Esta opción está solamente activa si hemos ejecutado el análisis LALR(1) anteriormente, y muestra las estructuras descritas en el punto anterior.

### 3.8. Menú Simular

Con este menú se puede simular entradas para una gramática determinada.

#### 3.8.1. Simular entrada LL(1)

Esta opción permite, una vez que se ha comprobado que la gramática actual es del tipo LL(1), simular el analizador sintáctico tipo LL(1) asociado a la gramática. Para ello el programa nos pedirá la entrada (formada por símbolos terminales) que queremos simular mediante una ventana como la que se muestra en la figura 28. En esta ventana introduciremos los símbolos separados por uno o varios espacios en blanco o tabuladores (para consultar el formato de la forma de frase a introducir nos referiremos al punto 3.10. **Formatos de entrada**).



**Figura 28: Petición de entrada del simulador**

Una vez que pulsemos el botón *Aceptar*, aparecerá la ventana de simulación donde se recogen las estructuras internas del simulador (pila de símbolos, producciones emitidas, entrada que falta por analizar y entrada original del simulador). Esta ventana se muestra en la figura 29.

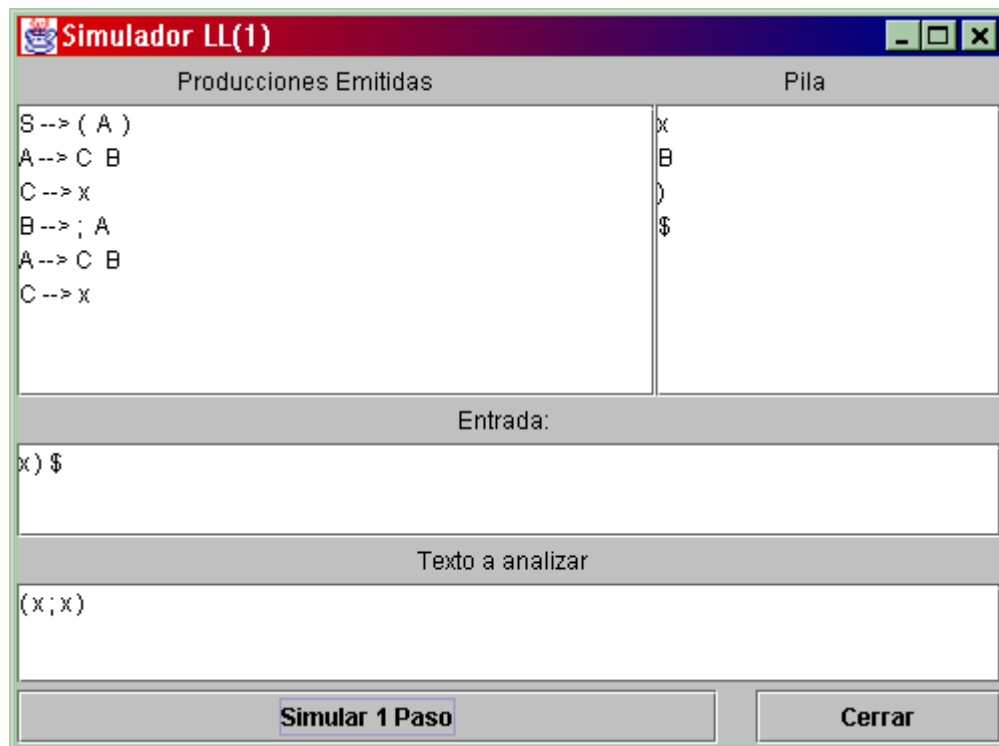


Figura 29: Ventana de simulación

En esta ventana, cada vez que pulsemos el botón *Simular 1 Paso*, se ejecutará un paso del proceso de simulación y se actualizarán las estructuras del analizador en pantalla. Además al comenzar la simulación aparecerá una nueva ventana donde se irá construyendo, también paso a paso, el árbol de sintaxis que corresponde a la entrada que se está simulando. Dicha ventana aparece en la figura 30.

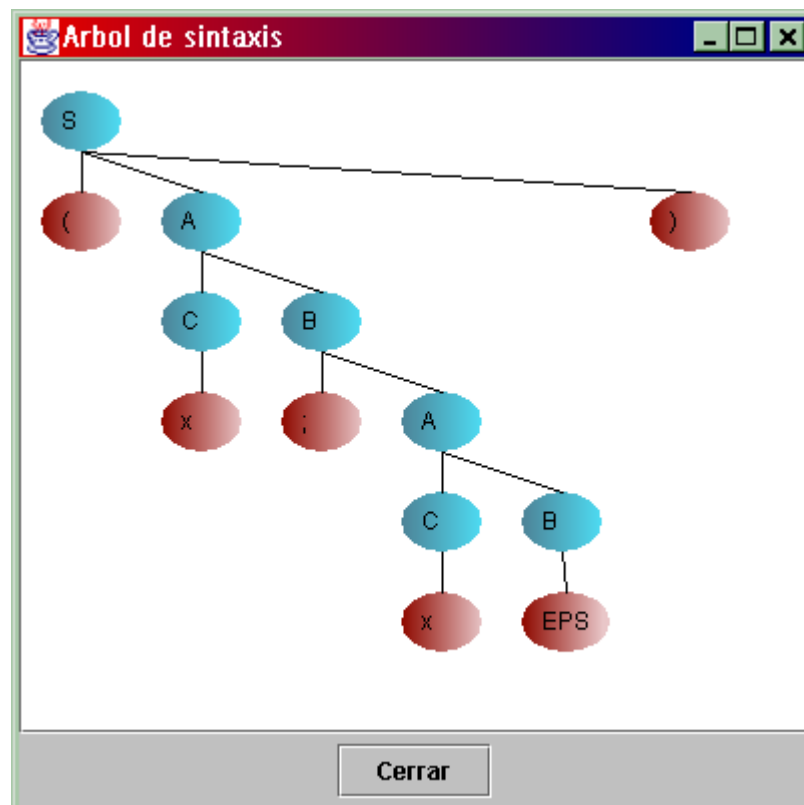


Figura 30: Árbol de sintaxis

### **3.8.2. Simular entrada SLR(1)**

El proceso para esta opción es idéntico al que se explica en el punto anterior. La única diferencia es que en este caso cambia el algoritmo de simulación.

### **3.8.3. Simular entrada LR(1)**

El proceso para esta opción es idéntico al que se explica en el 3.8.1. La única diferencia es que en este caso cambia el algoritmo de simulación.

### **3.8.4. Simular entrada LALR(1)**

El proceso para esta opción es idéntico al que se explica en el 3.8.1. La única diferencia es que en este caso cambia el algoritmo de simulación.

### **3.8.5. Comprobar entrada**

Esta opción permite la simulación directa de una entrada para una gramática que haya sido analizada satisfactoriamente para algún tipo de gramática. Tiene un campo donde se elige el tipo de analizador, otro campo para indicar si se quiere ver el árbol de sintaxis resultante de la simulación, y un área de texto para introducir la entrada a simular. En la figura 31 aparece la ventana de simulación.

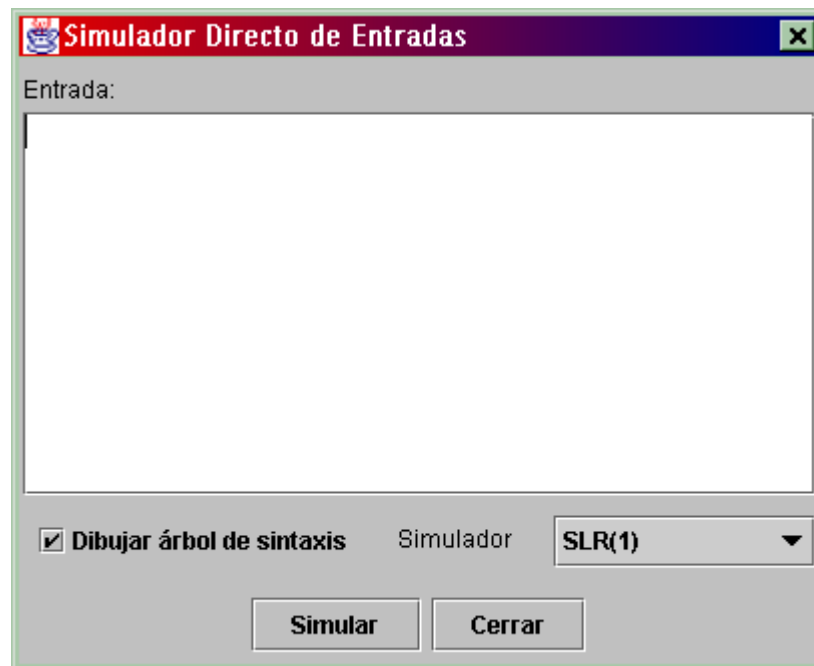


Figura 31: Ventana de simulación

## **3.9. Menú Ayuda**

### **3.9.1. Acerca de**

Contiene información acerca del proyecto y las personas que han intervenido en él. También ofrece una dirección de e-mail donde poder enviar los posibles errores encontrados.

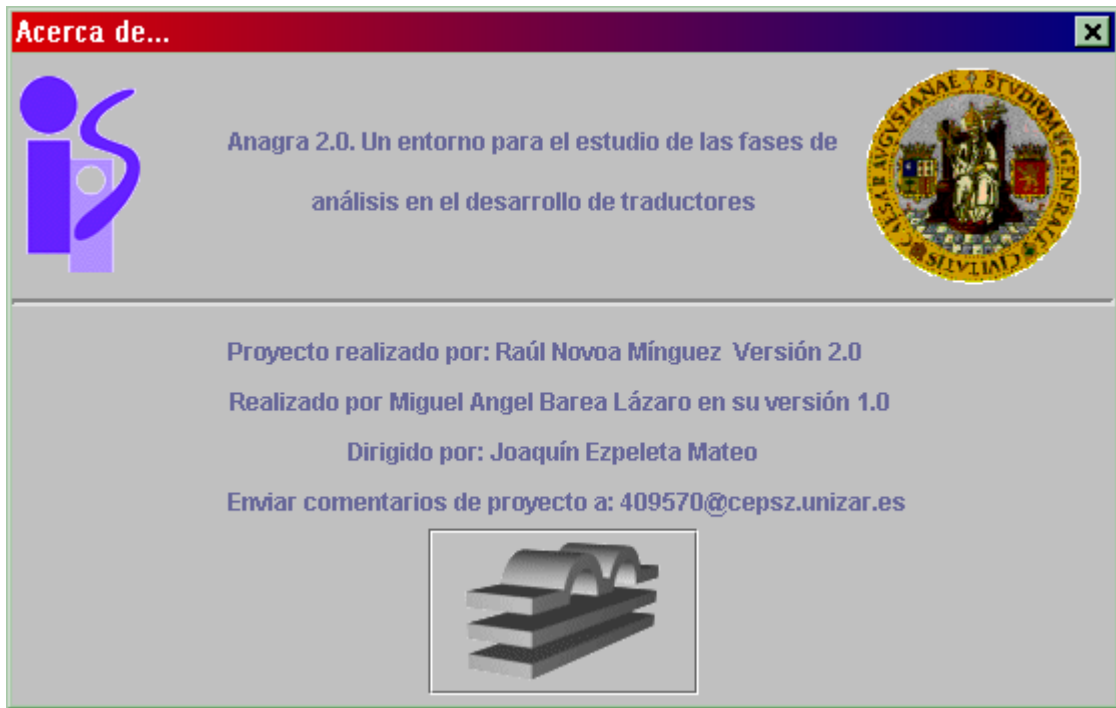


Figura 32: Diálogo Acerca de

### 3.10. Formatos de entrada

#### **3.10.1. Formato de los ficheros que almacenan las gramáticas**

El formato de los ficheros para introducir las gramáticas es el formato utilizado por YACC. Es decir, que podemos pasarle un fichero YACC y el programa lo reconocerá y obtendrá una gramática de él.

Hay que advertir que existen varias restricciones a lo indicado en el párrafo anterior.

1. Existen palabras reservadas a parte de las ya propias de YACC. A continuación se enumeran:
  - **EPS**
  - **Lambda**
  - **\$**
2. Los tokens terminales de la gramática podrán ser definidos al inicio del fichero con la declaración `%token...`, pero sólo se podrán definir de esta manera identificadores de texto (caracteres, números), esto es, si queremos tener un token que sea el signo mas '+' ó un paréntesis '(', etc.. deberemos definirlo dentro de las propias producciones como un símbolo literal (entre comillas simples). Si declaramos estos tokens en `%token` el programa dará un error en la gramática.

Cuando leamos una gramática desde un fichero, en la pantalla principal aparecerá el contenido íntegro del fichero. Cuando editemos la gramática o apliquemos alguna transformación sobre ella, en la pantalla principal solo aparecerá información acerca de las producciones y definición de tokens. Esta información será la que se almacene en un fichero de disco en caso de que seleccionemos dicha opción del menú.

#### **3.10.2. Formato de las formas de frase que se introducen en la aplicación**

**Forma de frase que se introduce para obtener el conjunto PRI.**

En este caso, se introducirán los símbolos de la gramática separados por uno o más espacios en blanco o tabuladores. Las mismas palabras reservadas que tenemos para los ficheros donde se almacenan las gramáticas, los tenemos aquí. Dichas palabras se especifican en el punto anterior. Ante cualquier error en la formación de la forma de frase, ANAGRA mostraría un mensaje de error indicándolo. Para ver posibles errores nos remitimos al punto **2.2.3. Cálculo del conjunto PRI de una forma de frase.**

**Entrada que simularemos con el analizador sintáctico asociado a las gramáticas.**

Similar al apartado anterior, pero puntualizando que aquí los símbolos de la gramática que utilizamos son sólo los terminales.

