

## Práctica 1: TABLA DE SÍMBOLOS

**Fecha de entrega: siguiendo indicaciones de tu profesor de prácticas.**

### Objetivo de la práctica

El objetivo de la práctica es la programación de la tabla de símbolos del compilador que se desarrollará durante el curso. Junto con la tabla de símbolos, también se codificará un programa de prueba que permita comprobar el buen funcionamiento de la tabla de símbolos programada.

### Desarrollo de la práctica

El esquema de desarrollo de esta práctica se mantendrá en la mayoría de las demás prácticas del curso. En primer lugar el alumno desarrollará en C el código concreto de la práctica, en este caso, la tabla de símbolos.

En segundo lugar, el alumno desarrollará un programa de prueba que permitirá comprobar el correcto funcionamiento del código desarrollado, en este caso la tabla de símbolos. Las especificaciones de este programa de prueba se detallarán en el enunciado de la práctica.

Por último, de manera opcional en función de la temática de la práctica, el alumno dispondrá de un conjunto de casos de prueba que le permitirán comprobar parcialmente el funcionamiento de su código.

### Codificación de una librería para la tabla de símbolos

La tabla de símbolos del compilador que se desarrolla durante el curso debe proporcionar la gestión correcta de ámbitos. Dadas las características del lenguaje ALFA, se puede afirmar que:

- Cuando el compilador se encuentra compilando código fuera de una función, solamente existe un ámbito (el global).
- Cuando el compilador se encuentra compilando una función, existen dos ámbitos abiertos (el ámbito global y el ámbito local de la función).

Es decir, en un instante dado de la compilación, puede haber como máximo dos ámbitos abiertos.

Una posible solución para la gestión de ámbitos en ALFA, es disponer de dos tablas, una para almacenar los identificadores del ámbito global, y otra para el almacenamiento de los identificadores locales de las funciones.

Para obtener un tiempo de compilación lo más bajo posible, es habitual el uso de tablas hash para la implementación de las tablas de símbolos de los compiladores. El alumno ya conoce este tipo de datos, los algoritmos para su gestión y su rendimiento.

El alumno debe escribir los ficheros y módulos C que considere necesarios para la definición e implementación de una tabla hash. Posteriormente, la tabla de símbolos del compilador consistirá en la combinación de dos de estas tablas hash.

La información que se almacena en la tabla de símbolos para cada identificador del programa se puede encontrar en la documentación correspondiente a las sesiones de trabajo de esta práctica.

## Codificación de un programa de prueba

Una vez desarrollado el código correspondiente a la tabla hash, el alumno escribirá en C un programa de prueba que permita simular el funcionamiento de la tabla de símbolos del compilador haciendo uso de dos tablas hash, una para el ámbito global, y otra para el local.

El objetivo es desarrollar un programa que permita comprobar el funcionamiento correcto de la tabla de símbolos. Esta corrección hace referencia a tres aspectos:

- Gestión de ámbitos
- Inserción de elementos
- Búsqueda de elementos

El programa leerá de un fichero de entrada operaciones de apertura/cierre de ámbitos y de inserción/búsqueda de elementos, realizará esas operaciones y escribirá en un fichero de salida el resultado de realizar cada una de dichas operaciones.

Se supone que:

- La tabla de símbolos está formada por dos tablas hash, una en la que se almacenan los identificadores de ámbito global (variables globales y nombres de funciones), que a partir de ahora se denominará tabla global y otra para los identificadores de ámbito local (variables locales de la función y parámetros de la función), que se denominará tabla local.
- Inicialmente está abierto el ámbito global, es decir, cualquier inserción o búsqueda se debe realizar en la tabla de ese ámbito (la tabla global).
- La apertura de un nuevo ámbito (correspondiente a una función) implica varias acciones:
  - En primer lugar se debe insertar el nombre del nuevo ámbito en la tabla global.
  - Seguidamente se debe abrir el ámbito local.
  - Por último se debe insertar el nombre del nuevo ámbito en su propia tabla local.
- El cierre de un ámbito implica el “vaciado” de la tabla local.
- Una operación de inserción termina con éxito si el identificador que se quiere insertar no existe en el ámbito actual.
- Una operación de búsqueda termina con éxito si el identificador se encuentra en alguno de los ámbitos abiertos (siempre se busca primero en la tabla local).

- Las operaciones de inserción utilizan dos parámetros, el identificador del elemento que se inserta y un número mayor o igual que 0, que se puede almacenar en cualquiera de los campos, por ejemplo en el campo “tipo”.
- Las operaciones de búsqueda que terminan con éxito recuperan el valor asociado al identificador (el número almacenado en el campo “tipo”).

Este programa de prueba se invocará con dos parámetros:

- El primer parámetro es el nombre del fichero de entrada.
- El segundo parámetro es el nombre del fichero de salida.

La descripción de los ficheros de entrada/salida se muestra a continuación.

### Descripción del fichero de entrada

El formato del fichero de entrada está diseñado para permitir probar el funcionamiento correcto de las operaciones de inserción y búsqueda en la tabla de símbolos junto con la correcta gestión de los ámbitos de visibilidad de los identificadores.

El fichero de entrada consiste en un conjunto de líneas, cada una de las cuales representa una acción sobre la tabla de símbolos. Estas acciones podrán ser:

- Inserción de un elemento
- Búsqueda de un elemento
- Apertura de un ámbito
- Cierre de un ámbito

Cada línea del fichero de entrada indica una acción sobre la tabla de símbolos según se indica en la siguiente lista. Se puede observar que cada línea contiene una cadena de caracteres seguida opcionalmente de un número entero (separados por un tabulador):

- **<identificador ámbito> <entero menor que -1>**. Significa que comienza un nuevo ámbito con identificador **<identificador ámbito>**. El programa deberá abrir un nuevo ámbito con todas las acciones que esto implica (descritas anteriormente).
- **cierre -999**. Significa que se cierra el ámbito activo.
- **<identificador> <entero mayor o igual que 0>**. Significa que se intentará insertar el identificador **<identificador>** (y se le asociará el valor **<entero mayor o igual que 0>**).
- **<identificador>**. Significa que se busca en la tabla el identificador **<identificador>**.

En la tabla siguiente se muestra un ejemplo del funcionamiento del programa de prueba sobre un posible fichero de entrada. En la primera columna aparecen las líneas del fichero de entrada. En la segunda columna se muestran las acciones que el programa de prueba realiza sobre la tabla de símbolos para cada línea del fichero de entrada:

<b>Línea del fichero de entrada</b>	<b>Acción sobre la tabla de símbolos</b>
uno 1	Intento de inserción en la tabla global del elemento (uno;1) La inserción terminaría con éxito porque el elemento no está
dos 2	Intento de inserción en la tabla global del elemento (dos;2) La inserción terminaría con éxito porque el elemento no está
uno 1	Intento de inserción en la tabla global del elemento (uno;1) La inserción terminaría con fallo porque el elemento ya está
dos 2	Intento de inserción en la tabla global del elemento (dos;2) La inserción terminaría con fallo porque el elemento ya está
uno	Búsqueda en la tabla del elemento (uno) La búsqueda terminaría con éxito porque el elemento está
dos	Búsqueda en la tabla del elemento (dos) La búsqueda terminaría con éxito porque el elemento está
cuatro	Búsqueda en la tabla del elemento (cuatro) La búsqueda terminaría con fallo porque el elemento no está
cinco	Búsqueda en la tabla del elemento (cinco) La búsqueda terminaría con fallo porque el elemento no está
funcion1 -10	Apertura de ámbito (funcion1) La apertura terminaría con éxito porque en la tabla global no existe. Se inserta el elemento en la tabla global, se inicializa la tabla local y se inserta el elemento en la tabla local
uno 10	Intento de inserción en la tabla local del elemento (uno;10) La inserción terminaría con éxito porque el elemento no está
uno 10	Intento de inserción en la tabla local del elemento (uno;10) La inserción terminaría con fallo porque el elemento ya está
funcion1 100	Intento de inserción en la tabla local del elemento (funcion1;100) La inserción terminaría con fallo porque el elemento ya está
tres 3	Intento de inserción en la tabla local del elemento (tres;3) La inserción terminaría con éxito porque el elemento no está
uno	Búsqueda en la tabla del elemento (uno) La búsqueda terminaría con éxito porque el elemento está
funcion1	Búsqueda en la tabla del elemento (funcion1) La búsqueda terminaría con éxito porque el elemento está
tres	Búsqueda en la tabla del elemento (tres) La búsqueda terminaría con éxito porque el elemento está
dos	Búsqueda en la tabla del elemento (dos)

	La búsqueda terminaría con éxito porque el elemento está
cuatro	Búsqueda en la tabla del elemento (cuatro) La búsqueda terminaría con fallo porque el elemento no está
cierre -999	Se cierra el ámbito local Se “vacía” la tabla local. La operación termina con éxito
funcion1	Búsqueda en la tabla del elemento (funcion1) La búsqueda terminaría con éxito porque el elemento está

El objetivo del programa de prueba es comprobar que las tablas hash funcionan de manera correcta y que se gestionan adecuadamente los ámbitos en cuanto a las operaciones de inserción y búsqueda. Queda fuera del alcance del programa de prueba controlar que en el fichero de entrada aparezcan, por ejemplo, operaciones sucesivas de apertura de ámbitos ya que esta situación es imposible en ALFA y por lo tanto no aparecerá nunca en el fichero de entrada. Tampoco aparecerán en el fichero de entrada cierres de ámbito cuando no se hayan abierto previamente.

## Descripción del fichero de salida

En el fichero de salida se deja constancia del resultado de cada petición solicitada en el fichero de entrada.

En el fichero de salida se tiene que poder distinguir entre las cuatro situaciones siguientes:

- **Inserción con éxito:** inserción de identificadores que no estén en el ámbito actual.
- **Inserción sin éxito:** inserción de identificadores que ya están en el ámbito actual.
- **Búsqueda con éxito:** búsqueda de identificadores que sí están en alguno de los ámbitos abiertos.
- **Búsqueda sin éxito:** búsqueda de identificadores que no están en ninguno de los ámbitos abiertos.

Cada línea del fichero de salida muestra el resultado de la acción solicitada en la correspondiente línea del fichero de entrada de acuerdo a los siguientes formatos:

- **<identificador>** para indicar que el identificador **<identificador>** se ha insertado con éxito en la tabla ya que no estaba previamente.
- **-1 <identificador>** para indicar que se ha solicitado la inserción de **<identificador>** pero no se ha podido terminar con éxito ya que el identificador ya estaba en la tabla fruto de una inserción previa.
- **<identificador> <entero>** para indicar que se ha solicitado la búsqueda del identificador **<identificador>** y se ha terminado con éxito puesto que previamente se había realizado su inserción con el valor asociado **<entero>**.
- **<identificador> -1** para indicar que se ha solicitado la búsqueda del identificador **<identificador>** pero no se ha podido completar con éxito ya que no ha sido insertado en la tabla.

- **cierre** para indicar que se ha cerrado el ámbito local.

El programa de prueba, para cada línea del fichero de entrada genera una nueva línea en el fichero de salida atendiendo a las reglas expuestas anteriormente. En la tabla siguiente se muestran juntos los ficheros de entrada y de salida correspondientes al ejemplo anterior.

<b>Línea del fichero de entrada</b>	<b>Línea del fichero de salida</b>	<b>Acción</b>
uno 1	uno	Inserción con éxito
dos 2	dos	Inserción con éxito
uno 1	-1 uno	Fallo de inserción
dos 2	-1 dos	Fallo de inserción
uno	uno 1	Búsqueda con éxito
dos	dos 2	Búsqueda con éxito
cuatro	cuatro -1	Fallo de búsqueda
cinco	cinco -1	Fallo de búsqueda
funcion1 -10	funcion1	Apertura de ámbito local
uno 10	uno	Inserción con éxito
uno 10	-1 uno	Fallo de inserción
funcion1 100	-1 funcion1	Fallo de inserción
tres 3	tres	Inserción con éxito
uno	uno 10	Búsqueda con éxito
funcion1	funcion1 -10	Búsqueda con éxito
tres	tres 3	Búsqueda con éxito
dos	dos 2	Búsqueda con éxito
cuatro	cuatro -1	Fallo de búsqueda
cierre -999	cierre	Cierre de ámbito
funcion1	funcion1 -10	Búsqueda con éxito

## Entrega de la práctica

El alumno entregará a través de moodle un fichero comprimido (.zip) que deberá cumplir los siguientes requisitos:

- Deberá contener todos los fuentes (ficheros .h y .c) necesarios para resolver el enunciado propuesto.
- Deberá contener un fichero Makefile compatible con la herramienta make que para el objetivo **all** genere el ejecutable de nombre **prueba\_tabla**.
- El nombre del fichero zip contendrá la identificación clara de los autores (ya sea con su número de grupo o con los apellidos de los integrantes del mismo)