

Proyecto Programado #2 – Valor 15%

Objetivos

- Introducir al estudiante en el manejo de tablas hash
- Describir el uso de diversas API para creación y gestión de tablas hash
- Implementar técnicas de programación asociadas al uso de APIs
- Implementar una tabla de símbolos

Instrucciones

Podemos definir una *tabla de símbolos* como una estructura de datos que asocia un valor con una llave. En términos generales, soporta las siguientes operaciones:

- Buscar que un nombre o elemento particular se encuentre en la tabla
- Obtener los atributos del elemento
- Modificar los atributos del elemento
- Insertar un nuevo elemento con sus atributos
- Borrar un elemento y sus atributos

Los compiladores, por ejemplo, utilizan las tablas de símbolos para almacenar información acerca de los elementos del código fuente del programa tales como nombres de variables y su tipo, nombres de funciones y nombre y tipo de parámetros así como el tipo de retorno, namespaces, los alcances, etc.

El proyecto consistirá en crear un pequeño analizador semántico de un archivo de código fuente, y específicamente se encargará de obtener el nombre y tipo de datos de una variable, funciones, y su alcance. El flujo del programa será el siguiente:



- 1) El programa deberá leer el código fuente (puede ser un archivo .txt)
- 2) A partir del código fuente, se deberá crear la tabla de símbolos para las funciones y variables. La tabla de símbolos se deberá crear a partir de funciones hash
- 3) La salida deberá mostrar en pantalla si el código fuente es correcto, o en caso contrario, deberá mostrar los errores (asignación de tipo incorrecta, nombre de variable no definido, etc.) así como el número de línea en donde se encontró el error

Características del Programa

- Puede asumir que solamente se van a trabajar cuatro tipos de datos en el código fuente: void, int, float y string. Cualquier otra declaración se tomará como tipo de datos no válido.
- Se asume que el código fuente solamente tiene declaraciones de tipos, funciones y el cuerpo de funciones, y las palabras reservadas serán "if" y "while". No existe una función main(), y además, en el alcance de las funciones, los "if" y los "while", se delimitará mediante llaves { }.
- El código fuente solamente contendrá: declaraciones, asignaciones, funciones y sentencias condicionales (if) y (while), y la sentencia 'return' para el valor de retorno de las funciones.

Ejemplo de código fuente correcto en el archivo .txt:

```
int x = 40;  
  
void funcion(float v, string n){  
    if (v > 0.0){  
        n = "Mayor"  
        x = x + 5  
    }
```

Observe que 'x' tiene alcance global, por lo que es válida dentro del contexto de la función. Las variables 'v' y 'n' también son válidas ya que están definidas como parámetros para la función.

Observe ahora un ejemplo de código fuente incorrecto, y los respectivos errores:

```
x = 40;  
  
int funcion(float v, string n){  
    if (v > 0.0){  
        n = "Mayor"  
        x = x + 5  
    }  
    return n
```

Error – Línea 1: 'x' no está declarado

Error – Línea 6: 'x' no está declarado

Error – Línea 8: valor de retorno no coincide con la declaración de 'funcion'

Observaciones

- La tarea debe ser realizada en grupos de tres personas como máximo.
- Se debe entregar el proyecto con el código fuente, ejecutables y librerías necesarias para su ejecución así como cualquier otro archivo que se requiera
- La fecha límite para la entrega del proyecto es el día Viernes 15 de noviembre de 2019 mediante el aula virtual
- En caso de copia (dos proyectos o más con mucha similitud que lo demuestren) o plagio (códigos descargados de Internet, libros o cualquier otro material), la nota de la tarea es 0

- No deben realizar documentación escrita de éste proyecto mas se tomará en cuenta la documentación del código como un rubro dentro de la calificación del mismo