

INFORME PRIMER PARCIAL - ALGORITMOS Y
ESTRUCTURAS DE DATOS 2022
FACULTAD DE CIENCIAS EXACTAS, FÍSICAS Y
NATURALES - UNC

Integrantes - Grupo 6:

Bengoechea, Matias

Gil Cernich, Manuel

Garbovetzky, Tomas

Consigna.

Desarrollar una máquina que ejecute un programa almacenado como una secuencia de instrucciones. Las instrucciones están escritas en un lenguaje de programación propio, con 5 tipos de instrucciones.

Ejemplo:

1. INT a
2. INT b
3. a = 4
4. b = 1
5. BOOL c
6. C = a<1
7. IF (c) THEN JUMP 12
8. b = b*a
9. a = a-1
10. c = a<1
11. JUMP 7
12. SHOW b

Desarrollo

Para capturar las instrucciones desde un archivo de texto, se creó el mismo en la misma carpeta que el código, con la sintaxis correcta, y se utilizó el método llamado "getline(string)" que captura línea a línea y las envía a una lista contenida dentro del método genérico Instrucción.

Luego dentro de un while que corta cuando la lista está vacía, con un switch buscamos palabras clave como por ejemplo INT, BOOL, IF, etc. para determinar las acciones a realizar.

La consigna planteaba crear una superclase instrucción de la cual heredarán las demás subclases específicas de cada tipo de instrucción. Nosotros entendíamos la lógica del programa y lo que debería de hacer cada clase, pero a la hora de implementarlo tuvimos problemas que no supimos cómo resolver. Uno de los inconvenientes fue que al pasar la lista de instrucciones por parámetro al método ejecutar y realizar la iteración al siguiente nodo, ese cambio no se ve reflejado en el main. Un problema similar surge cuando necesitamos recuperar y asignar un valor a un nodo de la lista de variables que ya estaba declarada previamente, el cambio en su valor no se ve reflejado. Dado esto, intentamos resolver el problema de otra manera, sin las clases. Para facilitarnos luego la implementación de clases, pero asimismo no pudimos realizarlo por problemas de implementación y tiempo.

A continuación se explica brevemente lo que hace el programa construido sin las clases.

Captura de instrucciones desde el archivo de texto y adición a la lista de instrucciones

Para leer las instrucciones del archivo se utilizó el método de la clase string llamado "getline" que captura cada línea y las envía a un string, y luego un bucle que las agrega a la lista de instrucciones. Para agregar nodos a la lista de instrucciones, se creó el método

“addinstruccion” (también hay un método para añadir variables, “addvar”, pero tiene un funcionamiento similar al código original de clases) que lo que hace es colocar los nodos (instrucciones) de forma “ordenada”, es decir, el nodo que se agrega siempre queda ubicado en la última posición de la lista (apuntando a NULL). De esta manera, nos aseguramos de que el contenido de cada nodo se lea en el mismo orden en el que está escrito el código fuente.

Ejecución de las instrucciones

El recorrido de los distintos nodos de la lista de instrucciones se realiza mediante un bucle while que recorre nodo a nodo dicha lista, mientras ésta tenga elementos (o sea, mientras sea distinta de vacía). Al mismo tiempo, se van contando el número de líneas leídas en una variable de tipo entero, que luego es utilizada para la secuencia “JUMP”. En el interior de este bucle se encuentra un operador switch, el que, de acuerdo al valor entero que es devuelto por la función BuscarPalabra, realiza las tareas que correspondan (inicialización, asignación, impresión y salto de líneas).