

CC1001-7 Auxiliar Extra 2 – Examen

Daniel Calderon – Sebastián Fehlandt

8 de Junio de 2009

Considere las siguientes clases:

TreeNode

<code>public TreeNode()</code>	Construye un nodo nulo
<code>public TreeNode(Object object)</code>	Construye árbol-hoja (hijos nulos)
<code>public TreeNode(Object object,TreeNode left,TreeNode right)</code>	Construye un arbol con información object y arboles hijos left y right.

StackNode

<code>public StackNode()</code>	
<code>public StackNode(Object x,StackNode next)</code>	Construye un Stack a partir de un objeto y un otro Stack. Deja el objeto arriba.
<code>public boolean empty()</code>	V o F si el Stack está vacío
<code>public void push(Object object)</code>	Pone el objeto en el Stack
<code>public Object pop()</code>	Saca el primer objeto del Stack

Tabla

<code>public Tabla(int N)</code>	Crea una tabla de máximo N elementos
<code>public void agregar(char var,double value) throws TablaLLena</code>	Agrega un registro a la tabla
<code>public double valueOf(char var) throws NoExiste</code>	Consulta el valor de var en la tabla.

Problemas

- a) Utilizando una pila y un árbol binario (clases anteriores), escriba una función que construya un árbol de operaciones matemáticas a partir de una expresión algebraica en notación de polaca. Ejemplo: la expresión $(a+b)*(c-d/e)$, en notación polaca es "ab+cde/-*". Esto es, al recorrer el string, al encontrar un operador, este actúa sobre las 2 últimas variables o expresiones algebraicas resultantes de otras operaciones. El encabezamiento debe ser:

```
public static TreeNode buildTree(String in)
```

- b) Escriba el método toString de la clase TreeNode. Este método debe retornar un string con la representación de infijo (tradicional) de la expresión almacenada en el árbol. No se preocupe por los paréntesis, póngalos siempre por si acaso.

```
public String toString()
```

- c) Escriba una función que traduzca una expresión algebraica desde notación polaca a notación de infijo. Esto es, si el argumento es "ab+cde/-*", debe retornar $(a+b)*(c-d/e)$. El encabezamiento debe ser:

```
public static String toInfixNotation(String in)
```

- d) Cada línea del archivo "datos.txt" contiene una letra seguida por su valor numérico. Escriba una función que construya una tabla con los valores de cada variable. El encabezamiento debe ser:

```
public static Tabla buildTabla(String file)
```

- e) Escriba una función que evalúe una expresión algebraica almacenada en el árbol binario según los datos de la tabla. El encabezamiento debe ser:

```
public static double evaluar(TreeNode t,Tabla datos)
```

- f) Implemente la clase Tabla con un arreglo. Hint: implemente una clase Variable.
- g) (Propuesto) Implemente las clases TreeNode, StackNode.