

# Estructuras de Datos

## Escuela de Informática-UNT

### Tarea2: MINARIT

Prof. José M. Saavedra R.  
Prof. Teresa Bracamonte N.

Fecha de Entrega: 22 de marzo, 2010

## 1 Objetivo

El objetivo de esta tarea es aplicar diversos TADs en la solución de problemas concretos. En particular, se espera que el alumno aplique listas enlazadas, colas y pilas.

## 2 Descripción

La tarea consiste en implementar un intérprete aritmético (algo así como un mini Matlab) al que llamaremos MINARIT. MINARIT recibe interactivamente una expresión como la siguiente:

```
> X = 5+3*4;
```

El símbolo > es el prompt de MINARIT e indica que está listo para recibir una expresión. La expresión siempre es una asignación o una simple variable.

Formalmente, la sintaxis de una expresión en MINARIT está dada por la siguiente:

$$\begin{aligned} \langle \text{expMINARIT} \rangle &\rightarrow \langle \text{variable} \rangle = \langle \text{expresionAritmetica} \rangle [;] \\ &\quad \langle \text{variable} \rangle [;] \\ \langle \text{variable} \rangle &\rightarrow \langle \text{letra} \rangle \{ \langle \text{letra} \rangle \mid \langle \text{digito} \rangle \}^* \\ \langle \text{expresionAritmetica} \rangle &\rightarrow \langle \text{variable} \rangle \mid \\ &\quad \langle \text{numero} \rangle \mid \\ &\quad (\langle \text{expresionAritmetica} \rangle) \mid \\ &\quad \langle \text{expresionAritmetica} \rangle \langle \text{operador} \rangle \langle \text{expresionAritmetica} \rangle \\ \langle \text{numero} \rangle &\rightarrow \langle \text{digito} \rangle \{ \langle \text{digito} \rangle \}^* \langle \text{digito} \rangle \{ \langle \text{digito} \rangle \}^* \\ \langle \text{operador} \rangle &\rightarrow \mid \mid + \mid - \\ \langle \text{digito} \rangle &\rightarrow 0 \mid 1 \mid 2 \mid \dots \mid 9 \\ \langle \text{letra} \rangle &\rightarrow A \mid B \mid C \mid \dots \mid Z \mid a \mid b \mid \dots \mid z \end{aligned}$$

En la sintaxis anterior:

- $\rightarrow$  : Indica que la expresión de la izquierda se puede reemplazar por la de la derecha. Sirve para definir NO TERMINALES.
- $\{ \text{exp} \}$  : Indica que *exp* se puede repetir 0 ó más veces.

- $[exp]$ : En este caso  $exp$  es opcional.
- $|$  : Indica OR.
- $\langle exp \rangle$  : Esta expresión se denomina NO TERMINAL, y debe ser reemplazada por alguna de sus definiciones.
- El resto de símbolos se denominan TERMINALES. (ej, + ,., (, ),; etc.)

## 2.1 Ejemplos de expresiones válidas

```
var=5
  x=8;
  z=x+2*var*(x+1);
  x
  x;
```

## 2.2 Requerimientos

Su programa, debe permitir que interactivamente el usuario ingrese una expresión de MINIRIT, y realizar lo siguiente:

1. Evaluar si la expresión es sintácticamente correcta. (Sugerencia: Puede comprobar primero que los paréntesis están balanceados y luego implementar un diagrama de estados, como un autómata, para revisar la sintaxis). Si la expresión no es sintácticamente correcta mostrar un mensaje de error apropiado.  
**Nota:** Es posible que hayan varios espacios en blanco en la expresión. Estos deben ser ignorados.
2. Debe convertir la expresión aritmética (la que va a la derecha de =) de notación INFIJA a notación POSTFIJA. De modo que el problema de precedencia de operadores quede resuelto. (Sugerencia: Mientras revisa la sintaxis de la expresión descompóngala en sus *tokens* como número, variable, operador, etc.).
3. Evalúe la expresión en notación POSTFIJA y relaciónela con la variable correspondiente. Durante la evaluación, si la expresión no termina con un punto y coma entonces muestre el resultado en la pantalla. Por ejemplo:

```
> x = 5;
> y=x
  y = 5
```

Es posible que ocurran errores en tiempo de ejecución. Esto sucederá si una variable utilizada NO tiene un valor asociado. En este caso, usted debe mostrar el mensaje de error correspondiente y terminar. Ejemplo:

```
> x = 5;
> y = x+z;
Error la variable z no se ha definido
```

### 3 Especificación

MINIARIT debe permitir ingresar interactivamente una expresión. El programa termina cuando escriba FIN. Por ejemplo:

```
MINIARIT
> x = 2;
> var = 3 - x
var = 1
> x = 5;
> z = var*5+x;
> z
10
>FIN
MINIARIT ha terminado.
```

### 4 Entrega

Debe entregar un informe con el desarrollo de la tarea, incluyendo ejemplos de uso (un manual de usuario). Junto con el informe entregará un CD con el código fuente del programa. 3

### 5 Restricciones

- El trabajo se puede realizar de hasta 2 estudiantes por grupo.
- El lenguaje de programación debe ser C o C++.
- Trabajos idénticos serán anulados.
- Los trabajos se entregan al inicio de la clase según fecha indicada.
- No se aceptan atrasos.