

Trabajo Práctico – Árbol de expresiones aritméticas.

Integrantes:

- Gimenez Erik
- Assale Andrea

Compilación:

Simplemente se hace uso del archivo “Makefile” disponible en la raíz de la carpeta del proyecto y la herramienta “make” disponible en Linux o su equivalente en Windows, por ejemplo, “mingw32-make” con el compilador MinGW. Esto generará un ejecutable llamado: “arbol_aritmetico”.

Dificultades:

La dificultad principal era la extracción de los operandos y operadores de la expresión matemática introducida por el usuario. Lo resolví usando principalmente la función strtok()

`(char *strtok(char *s1, const char *s2))`

Para extraer operadores y operandos, y strstr():

`char *strstr(const char *haystack, const char *needle)`

para saber desde donde debía comenzar a extraer los operadores y operandos de la entrada del usuario.

Otra dificultad fue el hecho de crear en sí el árbol aritmético y comprobar que la cantidad de números pasados a los operandos coincidieran con la aridad del operador. Además, había que evitar una división por 0 mientras se construía el árbol.

Lo resolví usando una pila de árboles aritméticos. Llevo la cuenta de los árboles que se apilan y cuando el programa detecta un operador, comprueba si su aridad coincide con la cantidad de árboles ya apilados.

Para evitar la división por cero, simplemente antes de apilar un nuevo árbol cuya raíz es el operador de división “/”, evalúo el subárbol a la derecha y, si es 0, cancelo la operación de carga, informo del problema y libero la memoria solicitada por la función hasta el momento

Casos de prueba:

1)

arbol = cargar 5 0 /

evaluar arbol

salir

La primera entrada imprimirá por consola un mensaje avisando que no se puede dividir por cero.

La segunda entrada imprimirá por consola un mensaje avisando que no se encontró el alias "árbol", ya que al intentar dividir por 0, no se carga la expresión-

La tercera, saldrá del programa, no sin antes liberar toda la memoria requerida

2)

1arbol = cargar 5 0 /

salir

La primera entrada imprimirá por consola un mensaje avisando que el primer carácter del comando no puede ser un número

La segunda entrada, saldrá del bucle del programa, no sin antes liberar toda la memoria requerida

3)

1arbol = cargar 5 1 /

salir

La primera entrada imprimirá por consola un mensaje avisando que el primer carácter del comando no puede ser un número

La segunda entrada, saldrá del programa, no sin antes liberar toda la memoria requerida

4)

Arbol = carga 5 1 /

salir

La primera entrada imprimirá por consola un mensaje avisando que el formato de la entrada es inválido (la palabra "cargar" está mal escrita).

La segunda entrada, saldrá del programa, no sin antes liberar toda la memoria requerida

5)

Arbol ? carga 5 1 /

salir

Igual al caso anterior, solo que esta vez, el problema está en el signo "?", el cual debería ser un signo de igualdad (=).

6)

Arbol = carga 5 1 /

Imprimir Ar

Acá, la expresión con el alias "Arbol", si se crea, pero la segunda entrada del usuario intenta imprimir una expresión que no existe, así que se avisará por un mensaje de consola que ese alias no está asociado a ninguna expresión.

7)

```
Arbol = cargar 3 -- +
```

Esta entrada es errónea porque se intenta usar el operador de suma cuando hay un solo número para sumar (-3, ya que el operador "--" crea un único operando, -3)

8)

```
Arbol = cargar 3 -- 4 +
```

```
Arbol = cargar 3 -- 4 + 2 *
```

```
imprimir = cargar 3 -- 4 + 2 *
```

```
cargar = cargar 3 -- 4 + 2 *
```

```
evaluar = cargar 3 -- 4 + 2 *
```

```
salir = cargar 3 -- 4 + 2 *
```

```
Arbol2 = cargar 3 -- 4 + 2 *
```

```
imprimir Arbol
```

```
evaluar Arbol
```

```
imprimir Arbol2
```

```
evaluar Arbol2
```

```
salir
```

La primera entrada creará correctamente una nueva expresión

La segunda entrada no creará una nueva expresión porque se intenta usar un alias ya utilizado

Desde la tercera hasta la sexta entrada, el programa no creará una nueva expresión ya que se está intentando usar palabras reservadas como alias de una nueva expresión.

En la séptima entrada se crea una nueva expresión

Hasta el final, se imprimen y evalúan las expresiones creadas.