

***Universidad Tecnológica Nacional - Facultad Regional Buenos Aires***

**Sintaxis y Semántica de los Lenguajes**

**2023**

**Trabajo Práctico N°2 - Automatas**

**Curso:** K2006

**Profesor:** Roxana Leituz

**Ayudantes:** -

**Repositorio:**

**Fecha Estipulada de Entrega:** 26-8-2023

**Grupo N°3**

|  |  |
| --- | --- |
| **Alumno** | **Legajo** |
| Petri Gianluca | 1776990 |

**Consigna**

Todos los trabajos deben llevar carátula con todos los datos formales.

Se deben incluir impresiones de las pantallas mostrando el funcionamiento del mismo.

La entrega debe contener los archivos fuente y un pdf con las pantallas, instructivo y especificaciones acerca de las decisiones tomadas para la resolución del mismo.

**AUTOMATAS**

1) Dada una cadena que contenga varios números que pueden ser decimales, octales o hexadecimales, con o sin signo para el caso de los decimales, separados por el carácter ‘$’ , reconocer los tres grupos de constantes enteras, indicando si hubo un error léxico , en caso de ser correcto contar la cantidad de cada grupo. Debe diagramar y entregar el o los autómatas utilizados y las matrices de transición. La cadena debe ingresar por línea de comando o por archivo.

2) Debe realizar una función que reciba un carácter numérico y retorne un número entero.

3) Ingresar una cadena que represente una operación simple con enteros decimales y **obtener su resultado**, se debe operar con +,- ,/, \*. Ejemplo = 3+4\*8/2+3-5 = 29. Debe poder operar con cualquier número de operandos y operadores respetando la precedencia de los operadores aritméticos y **sin paréntesis**.

La cadena ingresada debe ser validada previamente preferentemente reutilizando las funciones del ejercicio 1.

Para poder realizar la operación los caracteres deben convertirse a números utilizando la función 2.

La cadena debe ingresar por línea de comando o por archivo.

**NOTA:** SI CONSIDERAN QUE REQUIEREN BUSCAR ALGUNA FUNCIÓN EN LA WEB QUE LOS AUXILIE CON LA RESOLUCIÓN DE LA PRECEDENCIA DE OPERADORES PUEDEN HACERLO PERO DEBEN CITAR LA FUENTE, E INCORPORAR UNA EXPLICACIÓN DEL PROCEDIMIENTO, LA COPIA DE LA WEB ES SOLO A ESTE EFECTO, NADA DEL TRABAJO PUEDE SER COPIADO, DEBE SER ORIGINAL, **SI SE DETECTA COPIA AUTOMATICAMENTE PIERDEN LA PROMOCION Y TENDRÁN QUE HACER UN TRABAJO EXTRA PARA APROBAR LA MATERIA**.

**El lenguaje de programación a utilizar debe ser C, no se aceptan trabajos en C++.**

**Para aprobar debe estar completo y funcionando correctamente, utilizar buenas prácticas de desarrollo e interfaces amigables. Este trabajo puede requerir coloquio al finalizar la cursada a criterio del docente.**

**Resolución**

1. Primero comienzo por presentar la BNF en C para las formas en las que puede venir representados mis valores numericos de distintas bases:

<constante entera> ->

<constante decimal> <sufijo entero>? |

<constante octal> <sufijo entero>? |

<constante hexadecimal> <sufijo entero>?

<constante decimal> ->

<dígito no cero> |

<constante decimal> <dígito>

<dígito no cero> -> uno de

1 2 3 4 5 6 7 8 9

<dígito> -> uno de

0 1 2 3 4 5 6 7 8 9

<constante octal> ->

0 |

<constante octal> <dígito octal>

<dígito octal> -> uno de

0 1 2 3 4 5 6 7

<constante hexadecimal> ->

0x <dígito hexadecimal> |

0X <dígito hexadecimal> |

<constante hexadecimal> <dígito hexadecimal>

<dígito hexadecimal> -> uno de

0 1 2 3 4 5 6 7 8 9 a b c d e f A B C D E F

<sufijo entero> ->

<sufijo "unsigned"> <sufijo "long">? |

<sufijo "long"> <sufijo "unsigned">?

<sufijo "unsigned"> -> uno de

u U

<sufijo "long"> -> uno de

l L

Con esta información planteo los autómatas correspondientes a los 3 tipos de números que puedo recibir:

**Autómata para números decimales con o sin signo:**

Dno

+,-

1

D

2+

0-

Dno

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **+** | **-** | **Dno** | **D** |
| **0-** | 1 | 1 | 2 | 3 |
| **1** | 3 | 3 | 2 | 3 |
| **2+** | 3 | 3 | 3 | 2 |
| **3** | 3 | 3 | 3 | 3 |

<+,-> = signos

<Dno> = 1 2 3 4 5 6 7 8 9

<D> = 0 1 2 3 4 5 6 7 8 9

**Autómata para números octales (deben comenzar con <0>):**

D

1

0

0-

|  |  |  |
| --- | --- | --- |
|  | **0** | **D** |
| **0-** | 1 | 2 |
| **1+** | 1 | 1 |
| **2** | 2 | 2 |

<0> = cero

<D> = 0 1 2 3 4 5 6 7

**Autómata para números hexadecimales (deben comenzar con <0x>):**

0

1

<D>

<prefijo>

2+

0-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **0** | **x** | **X** | **D** |
| **0-** | 1 | 3 | 3 | 3 |
| **1** | 3 | 2 | 2 | 3 |
| **2+** | 3 | 3 | 3 | 2 |
| **3** | 3 | 3 | 3 | 3 |

<prefijo> = x | X

<D> = 0 1 2 3 4 5 6 7 8 9 a b c d e f A B C D E F