



Trabajo Práctico N.º 4

En base a la gramática Sintáctica hecha en el TP2 y la implementación de la gramática léxica hecha en el TP3 armar un programa que usando la herramientas flex y bison reconozca el lenguaje de la calculadora.

En este TP queremos reconocer la sintaxis sin hacer realmente los cálculos ni reconocer los posibles errores semánticos.

Consideraciones:

- Tanto en flex como bison deben poner las directivas que eviten tener que agregar opciones adicionales al invocar las herramientas en la línea de comando
- La función main estará en su propio fuente main.c que llamará a yyparse.
- Se debe proveer un mecanismo de recuperación de errores sintácticos de modo que el programa no termine con el primer error.
- Para un mejor seguimiento pondremos en bison acciones semánticas que permitan ver a grandes rasgos que es lo que va analizando, para ello:
 - Luego de una declaración diremos si se declaró una variable, una variable con inicialización o una constante.
 - En cada uno de los casos de expresión diremos que operador o bien si fué un identificador o un número o si se llamó a una función (ver ejemplo de salida)
- ACHATAR GRAMATICA y usar las directivas necesarias para establecer precedencia y asociatividad.

Sugerencias:

- Utilice las directivas noinput y nounput para evitar warnings innecesarios

Se incluye el archivo entrada.txt que puede utilizar para hacer pruebas, y el archivo salida.txt que es la salida correspondiente a la entrada de prueba. Por supuesto el formato es solo un ejemplo, no hay necesidad de que lo haga exactamente igual, si que muestre la información que se pide de un modo razonable.

Entrega: Directorio compactado con los archivos: scanner.l, main.c y parser.y.

Si lo consideran necesario pueden entregar archivos adicionales, por ejemplo más fuentes o encabezados