**Sintaxis y Semántica de los Lenguajes**

**TPN4 - Compilador de Micro**

Integrantes - Grupo 7

| ALUMNO | Legajo |
| --- | --- |
| Giovani Quispe | 159978-1 |
| Guido Biotti | 171.434-0 |
| Santiago Hamamura | 203.717-8 |

Profesora Ing. Roxana Leituz

Curso K2006 - Sábados Turno mañana

2023

Enunciado

1) Compilar el Código entregado en clase.

2) Preparar 3 programas en micro: 1 correcto , 1 con error léxico , 1 con error sintáctico y semántico.

3) Analizar como funciona el análisis de cada uno de los errores expresando con sus palabras porque se produce el error, no relatando las líneas del código sino con lenguaje natural.

Lineamientos de entrega:

• Carátula con todos los integrantes

• Ejecutable

• Documento con

a. pantallas mostrando que el programa funciona

b. análisis de los errores solicitados

• Archivos anexos: se deben incluir los archivos que se utilicen en las pruebas

• Grupo: el mismo de todo el año

• Caso de copia: desaprueban la materia y deberán recuperar en marzo perdiendo por completo la posibilidad de promoción.

• El trabajo para darse por cumplido debe entregarse con todos los ítems desarrollados.

• Se entrega por campus una copia por grupo.

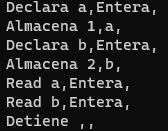
Introducción

Los 3 archivos de prueba con código Micro (.m) se ubican en la carpeta /MicroFiles. La carpeta /src contiene el código C (.c) fuente del compilador de Micro utilizado, junto a el binario ejecutable “microcompiler.exe”. La manera de invocarlo es por terminal:

microcompiler.exe <filename.m>

Programa correcto

Se enseña el código presente en “sinError.m” junto a la salida del compilador:



inicio

a:= 1;

b:= 2;

leer(a);

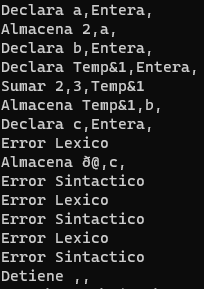
leer(b);

fin

Para este programa definimos 2 variables a que se declara y se le almacena el valor 1 y b que se la declara despues y se le almacena el valor 2. Despues de esto decidimos leerlas para que se vea el correcto funcionamiento del micro.

Error léxico

Se enseña el código presente en “lexico.m” junto a la salida del compilador:



inicio

a := 2;

b := 2+3;

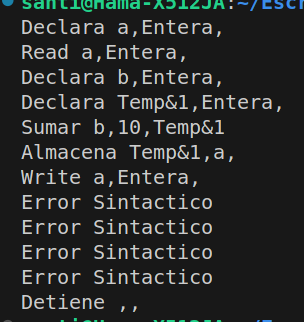
c := @#$$%%;

fin

Para este codigo inilcializamos 3 variables. La primera declarada es a y se le almacena el 2, la segunda es b y se declara una variable temporal para guardar en b la suma de 2 y 3 y por ultimo declaramos c que al ser un variable no entera nos genera un error lexico al declararla y luego al tratar de almacenarla genera mas errores sintacticos y lexicos para luego detener el programa.

Error sintáctico y semántico

Se enseña el código presente en “syntaxError.m” junto a la salida del compilador:



inicio

leer(a);

a := b + 10;

escribir(a := 123);

fin

Se comienza definiendo una variable ´a´ por I/O. Luego, se le asigna ´b+10´, definiendo implícitamente a la variable b cuyo valor no está definido. Como resultado, debería enseñarse un error de tipo semántico. Luego se pretende realizar una escritura, pero como parámetro se ingresa una asignación, propia de las sentencias y no de las expresiones que espera recibir escribir(), separadas por comas. Esto debería resultar en otro error semántico, además de los errores sintácticos que sí se imprimen por salida. No hay error léxico, pues todos los tokens pueden identificarse correctamente.