Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Lenguajes Formales y de Programación A
Ing. Luis Espino
Aux. Aylin Aroche
Vacaciones junio 2019

PRACTICA 1

Objetivos:

- Que el estudiante desarrolle un pensamiento lógico para la implementación de un analizador léxico.
- Que el estudiante desarrolle habilidades para posteriormente utilizarlos en el desarrollo de analizadores sintácticos.
- Que el estudiante aplique los conocimientos adquiridos en la clase y en el laboratorio.

Descripción:
La práctica consiste en realizar un diagrama para representar el pensum de una Facultad para cualquier Universidad que lo requiera, así como los cursos que dicha Facultad posee, para dicho diagrama se tiene un lenguaje definido, y a través de éste se pueden generar gráficas que muestren las relaciones entre los diferentes cursos, con sus dichos prerrequisitos y créditos. La aplicación será desarrollada en lenguaje de programación Java.

El gráfico propio del diagrama se realizará a partir del contenido del editor de diagrama, este editor se puede llenar directamente o a través de la carga de un archivo con extensión .psum, y los gráficos serán mostrados en una página HTML externa a la aplicación. En caso de que el archivo contenga errores, mostrará una página HTML con el detalle de estos.

Definición del lenguaje:
El lenguaje para generar los diagramas tendrá los siguientes elementos:

- **PENSUM:** Este bloque será el bloque principal y contendrá la información del diagrama: nombre de la facultad y cursos.
- **NOMBRE:** Este bloque tendrá el nombre de la facultad a la que están asociados los cursos.
- **CURSO:** Este bloque tendrá la información del curso: código del curso, nombre, prerrequisitos y los créditos que proporciona. Puede venir uno o más cursos dentro del bloque pensum. El bloque curso tiene los siguientes elementos:
    - **CODIGO:** El código del curso debe de ser un número entero mayor a cero.
    - **NOMBRE:** Define el nombre del curso, debe estar encerrado en comillas dobles.
    - **CREDITOS:** Define los créditos que proporciona el curso, debe ser un número entero mayor o igual a cero.
    - **PRERREQUISITOS:** Contiene un conjunto de números enteros mayores a cero, separados por coma, que representan los códigos de los cursos que el estudiante debe haber aprobado con anticipación para poder llevar dicho curso. Para el caso

en el que el curso no tenga ningún prerrequisito, el conjunto de códigos estará vacío.

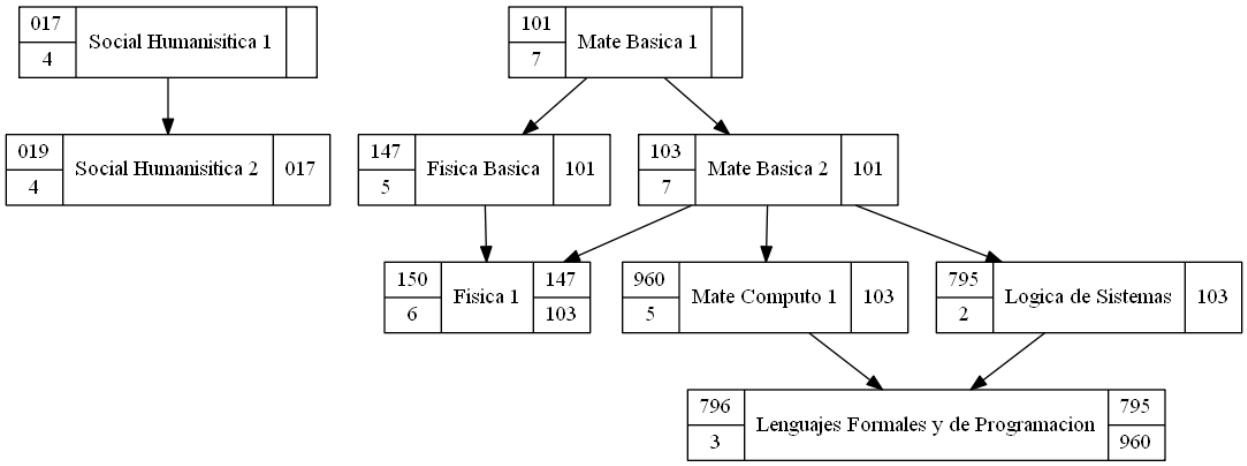
Para cada bloque PENSUM que se ingrese en el archivo de entrada, se generará una página HTML que mostrará el nombre de la facultad y el grafico correspondiente. Todos los nombres de los componentes deberán ir en minúscula.

Archivo de entrada ejemplo:

```
pensum: "Facultad de Ingeniería":{
        curso:{
                codigo:017;
                nombre:"Social Humanisitica 1";
                creditos: 4;
                prerrequisitos:{};
        }
        curso:{
                codigo:019;
                nombre:"Social Humanisitica 2";
                creditos: 4;
                prerrequisitos:{017};
        }
        curso:{
                codigo:101;
                nombre:"Mate Basica 1";
                creditos: 7;
                prerrequisitos:{};
        }
        curso:{
                codigo:103;
                nombre:"Mate Basica 2";
                creditos: 7;
                prerrequisitos:{101};
        }
        curso:{
                codigo:147;
                nombre:"Fisica Basica";
                creditos: 5;
                prerrequisitos:{101};
        }
        curso:{
                codigo:150;
                nombre:"Fisica 1";
                creditos: 6;
                prerrequisitos:{103,147};
        }
        curso:{
                codigo:960;
                nombre:"Mate Computo 1";
                creditos: 5;
                prerrequisitos:{103};
        }
        curso:{
                codigo:795;
```

```
                nombre:"Logica de Sistemas";
                creditos: 2;
                prerrequisitos:{103};
        }
        curso:{
                codigo:796;
                nombre:"Lenguajes Formales y de Programación";
                creditos: 3;
                prerrequisitos:{795,960};
        }
}
```

Al ejecutar las instrucciones anteriores en el editor, se generaría una página HTML que mostraría la siguiente imagen.



La página HTML debe contener el nombre del diagrama indicado en el archivo. Cada uno de los componentes muestra únicamente el texto, las flechas del grafo son generadas en función de los prerrequisitos de cada curso. El grafo debe tener la misma estructura mostrada en el ejemplo.

Aplicación:

La aplicación debe analizar el texto del editor, de encontrar errores léxicos, la aplicación debe informar al usuario y mostrar un reporte de errores en formato HTML, en este caso no debe generar la página de salida. En la tabla con el reporte de errores se deben mostrar los siguientes campos:

No.	Error	Descripción	Fila	Columna
1	%	Elemento léxico desconocido	3	4

De no encontrar error alguno, se debe generar la página HTML de salida correspondiente. La aplicación debe pintar dentro del editor cada uno de los lexemas, según su tipo, los colores se definen a continuación:

Tipo	Color
Palabras reservadas	Azul
Números	Rojo
Cadenas	Verde
Llaves y corchetes	Morado
Coma, punto y coma, dos puntos	Amarillo
Otros	Negro

Además de colorear los elementos léxicos y generar la página de salida HTML, se debe generar otra página HTML que contenga una Tabla de tokens con los siguientes campos:

No.	Token	Lexema	Tipo	Fila	Columna
1	5	curso	Reservada	2	1
2	10	“Texto”	Cadena	2	12
3	2	,	coma	2	11

Interfaz gráfica:

Componentes mínimos de la interfaz:

- **Editor de texto:** Debe ser un área de texto en el cual se puedan escribir nuevas instrucciones de entrada o bien cargar texto desde archivos con extensión .psum
- **Abrir:** La interfaz debe proveer la capacidad de abrir archivos con extensión .psum
- **Guardar**: La aplicación debe proveer la capacidad de guardar el texto contenido en el editor. Si no ha sido guardada nunca, debe proveer una opción para ingresar el nombre del nuevo archivo y la ubicación en la que se desea guardar.
- **Guardar como:** Está opción permite guardar el archivo de entrada con otro nombre, se debe preguntar el nombre del nuevo archivo.
- **Analizar:** Debe realizar el análisis léxico del lenguaje que se encuentra actualmente en el editor y generar la salida correspondiente.
- **Acerca de:** Debe desplegar una ventana con los datos del estudiante y del curso.
- **Salir:** Debe terminar la ejecución de la aplicación.

Entregables:

- Manual de Usuario
- Manual Técnico:
  - Debe incluirse el autómata finito determinista que se utilizó para la implementación del analizador léxico.
  - Plataforma de ejecución
  - Diagrama o diccionario de clases
  - Herramientas utilizadas
- Código fuente.
- Ejecutable de la práctica

Notas importantes:

- La práctica se debe desarrollar de forma individual.
- Esta práctica se deberá desarrollar en el lenguaje java.

- Graphviz es la única herramienta válida para generar los gráficos.
- El proceso de obtener tokens, se debe hacer a través de la implementación del autómata finito determinista desarrollado por el propio estudiante.
- La calificación se realizará ÚNICAMENTE con los archivos que el auxiliar provea en el momento de la calificación. Estos archivos no podrán ser modificados de ninguna manera.
- La calificación de la práctica será personal y durará como máximo 30 minutos, en un horario que posteriormente será establecido. Se debe tomar en cuenta que durante la calificación no podrán estar terceras personas alrededor, de lo contrario no se calificará la práctica.
- No se dará prórroga para la entrega de la práctica.
- Copia parcial o total de la práctica tendrá una nota de 0 puntos, y se notificará a la escuela de sistemas para que se apliquen las sanciones correspondientes.
- En el caso de no cumplir con alguna de las indicaciones antes mencionadas, NO se calificará el proyecto; por lo cual, se tendrá una nota de cero puntos.

Anexos

- Sitio oficial de Graphviz: http://www.graphviz.org/
- Tutorial de instalación de Graphviz: https://www.youtube.com/watch?v=JYAHwQ_tMG0

Fecha de entrega: 11 de Junio de 2019