

VACACIONES DE DICIEMBRE 2021



# MYCAREER-USAC

*Manual de Usuario*

**ELABORADO POR:**

Rodrigo Alejandro Hernández de León  
201900042

**LENGUAJES FORMALES Y  
DE PROGRAMACIÓN - N**

# Contenido

I.	Introducción.....	3
1.	Objetivo.....	3
2.	Requerimientos.....	3
II.	Opciones del Sistema.....	4
1.	Ingreso a la Aplicación .....	4
2.	Descripción del archivo .lfp.....	5
	nombre_de_red:.....	6
	crearcurso: .....	6
	consola: .....	7
	consolaln: .....	8
	cursosPorSemestre: .....	8
	cursoPorCodigo:.....	8
	cursoPorNombre: .....	9
	cursosPrerrequisitos: .....	9
	cursosPostrequisitos:.....	9
	generarRed:.....	10
3.	Cargar Archivo.....	10
4.	Escritura del lenguaje .lfp .....	11
5.	Analizar Texto .....	12
	nombre_de_red:.....	13
	crearcurso: .....	13
	consola: .....	13
	consolaln: .....	13
	cursosPorSemestre: .....	13
	cursoPorCodigo:.....	14

- cursoPorNombre: .....14
- cursosPrerrequisitos: .....15
- cursosPostrrrequisitos:.....15
- generarRed:.....16
- 6. Generar Reportes .....17
  - Reporte de Tokens: .....17
  - Reporte de Errores:.....17
  - Árbol de derivación:.....18
- III. Recomendaciones .....19

# I. Introducción

## 1. Objetivo

Otorgar el soporte para el uso adecuado del lenguaje a programar en la consola del programa donde tendrá control de su red curricular de estudios.

## 2. Requerimientos

- Computadora portátil o de escritorio.
- Mínimo 4GB de Memoria RAM.
- Cualquier sistema operativo que contenga un navegador web.
- Mínimo 250 GB de almacenamiento en el Disco Duro.
- Resolución gráfica mínimo 1024 x 768 pixeles.
- Tener instalado Python en la versión 3.9.2

## II. Opciones del Sistema

En las siguientes paginas se estará explicando el uso de las funcionalidades que tiene la aplicación:

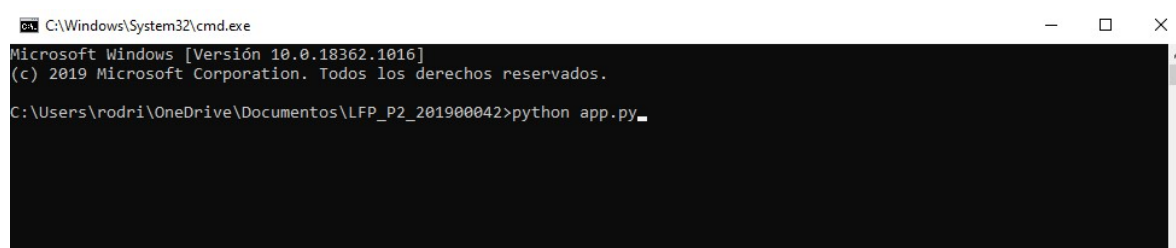
1. Ingreso a la Aplicación.
2. Descripción del archivo .lfp
3. Cargar Archivo.
4. Escritura del código .lfp
5. Analizar Texto.
6. Generar Reportes.

### 1.Ingreso a la Aplicación

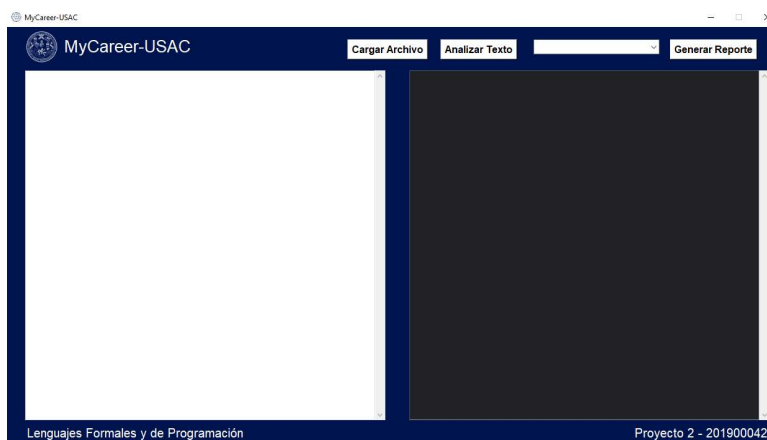
Ingresa a la carpeta del proyecto y escriba CMD donde se encuentra la ruta:



Presione enter y le aparecerá la consola de Windows, y escriba lo siguiente: `python app.py`




Presione enter y le mostrará la siguiente ventana:



## 2.Descripción del archivo .lfp

El contenido del archivo .lfp puede ser el siguiente:

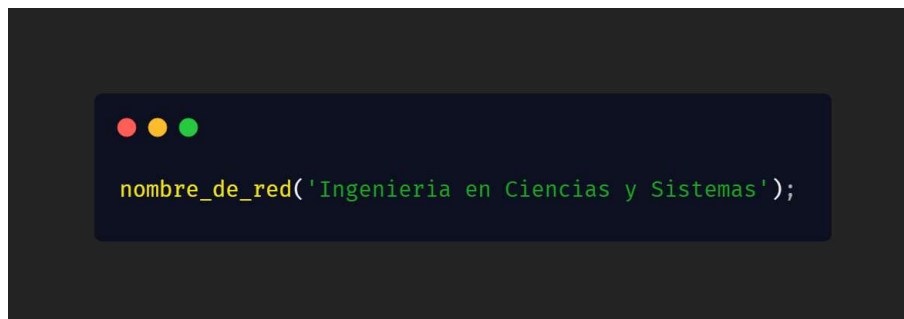


```
nombre_de_red('Nombre de la carrera');
crearcursos(No.semestre, codigo, 'Nombre del curso', []);
crearcursos(No.semestre, codigo, 'Nombre del curso', [Codigo]);
consola('texto');
consolaln('texto');
cursosPorSemestre(numero);
cursoPorCodigo(codigo);
cursoPorNombre('Nombre del curso');
cursosPrerrequisitos(codigo);
cursosPostrrrequisitos(codigo);
generarRed('Nombre del archivo');
```

Donde este lenguaje se compone con las siguientes funcionalidades:

### **nombre\_de\_red:**

Es la que permite configurar el nombre de la carrera o red de estudios con la que estará trabajando, un ejemplo de su estructura es la siguiente:

A screenshot of a code editor with a dark background. It shows a single line of code: `nombre_de_red('Ingenieria en Ciencias y Sistemas');`. The text is color-coded: `nombre_de_red` is in green, the opening parenthesis and the string are in yellow, and the closing parenthesis and semicolon are in green. Above the code line are three small colored circles (red, yellow, green) representing window controls.

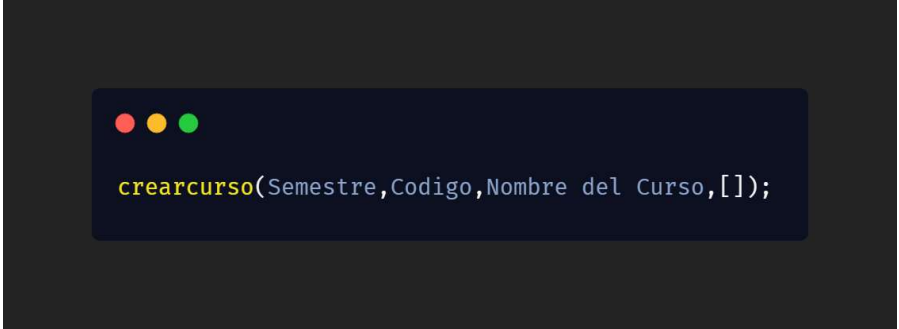
### **crearcurso:**

Permite crear muchos cursos que pertenecen a la red de estudios con la palabra reservada `crearcurso` seguida de los siguientes parámetros encerrados entre paréntesis y separados por comas, al finalizar la instrucción lleva punto y coma:

- **Semestre:**  
Numero de semestre a la cual pertenece el curso, es un número entero.
- **Código:**  
Código del curso que se esta agregando, es un numero entero.
- **Nombre del curso:**  
Nombre del curso que se esta agregando, es una cadena de texto.
- **Prerrequisitos:**  
Arreglo de códigos de cursos prerrequisitos para llevar el curso que se está agregando, se encierran entre corchetes y se separan por comas si hubiera más de un curso

prerrequisito, de no tener prerrequisito los corchetes estarán vacíos, estos códigos de cursos son números enteros.

Esta es la estructura de la instrucción:



```
crearcurso(Semestre,Codigo,Nombre del Curso,[]);
```


Un ejemplo de este sería:



```
crearcurso(1,150,'Fisica1',[103,147]);
```

### **consola:**

Imprime por consola el valor dado por la cadena, un ejemplo de su estructura es la siguiente:



```
consola('Hola mundo');
```

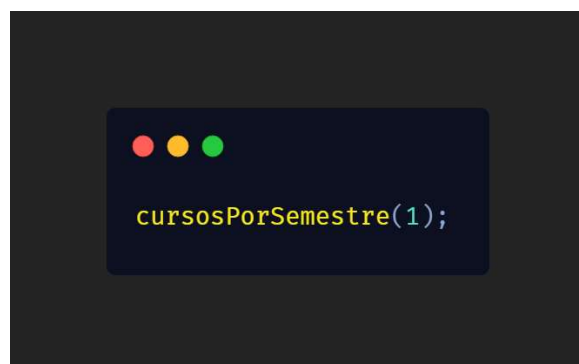


**consoleLn:**

Imprime en consola el valor dado por la cadena y realiza un salto de línea, y un ejemplo de su estructura es la siguiente:

**cursosPorSemestre:**

Imprime por consola la información de los cursos dados por el parámetro, en este caso imprimiría la información de todos los cursos de primer semestre.

**cursoPorCodigo:**

Imprime por consola la información del curso dado por el parámetro, un ejemplo es el siguiente.



**cursoPorNombre:**

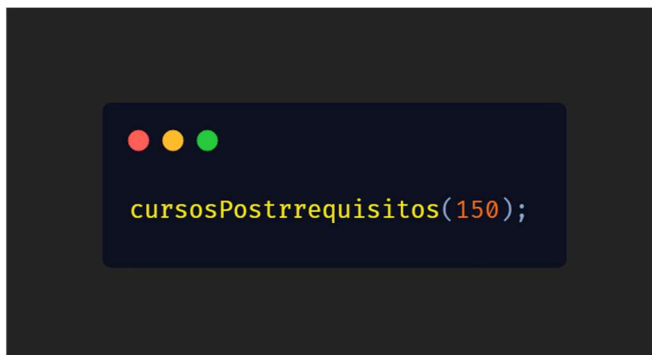
Imprime por consola la información del curso dado por el parámetro, un ejemplo de su estructura es el siguiente:

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The text `cursoPorNombre('Fisica 1');` is displayed in a light green monospace font.**cursosPrerrequisitos:**

Imprime por consola los cursos prerrequisitos del curso dado por el parámetro, un ejemplo de su estructura es la siguiente:

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The text `cursosPrerrequisitos(150);` is displayed in a light green monospace font.**cursosPostrrrequisitos:**

Imprime por consola los cursos postrrrequisitos del curso dado por el parámetro, un ejemplo de su estructura es la siguiente:

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The text `cursosPostrrrequisitos(150);` is displayed in a light green monospace font.

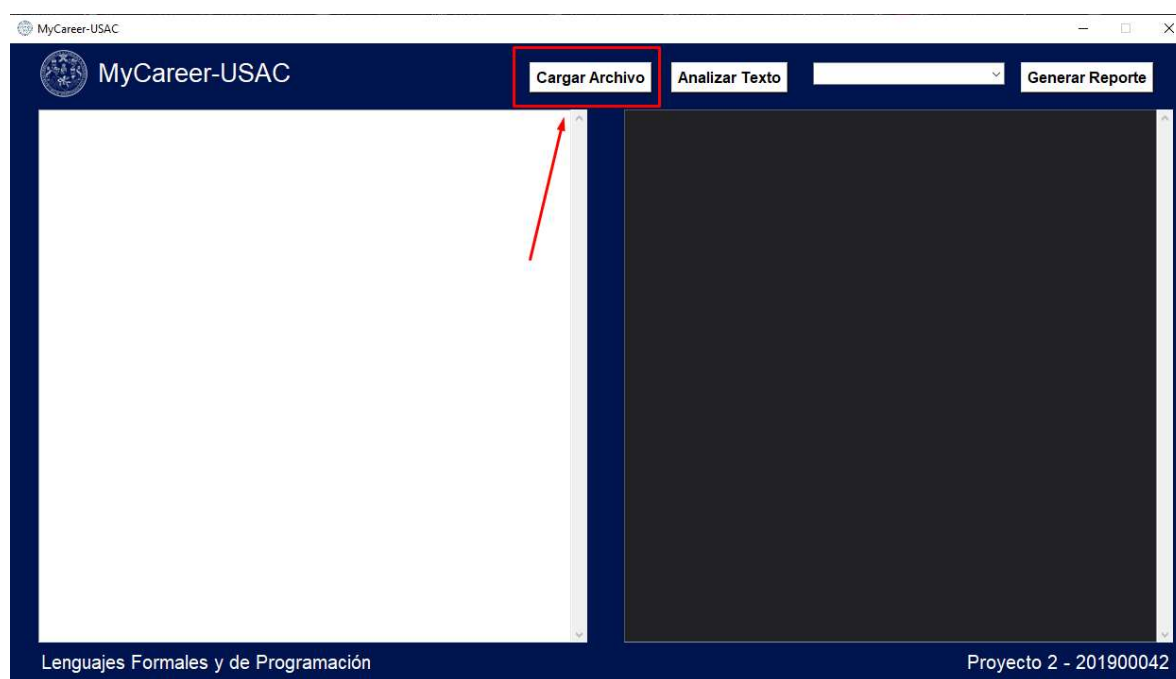
## generarRed:

Guarda el grafo de la red de estudios con el nombre dado en formato .png y .gv, un ejemplo de la estructura seria la siguiente:

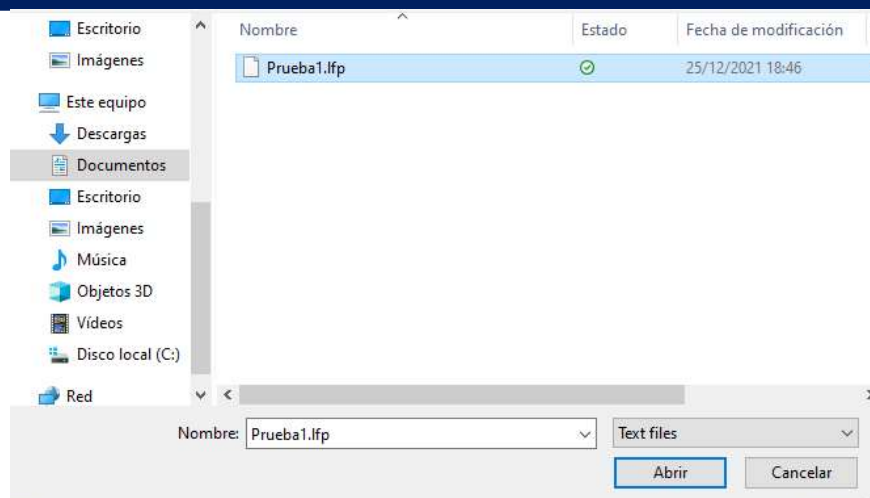


## 3. Cargar Archivo

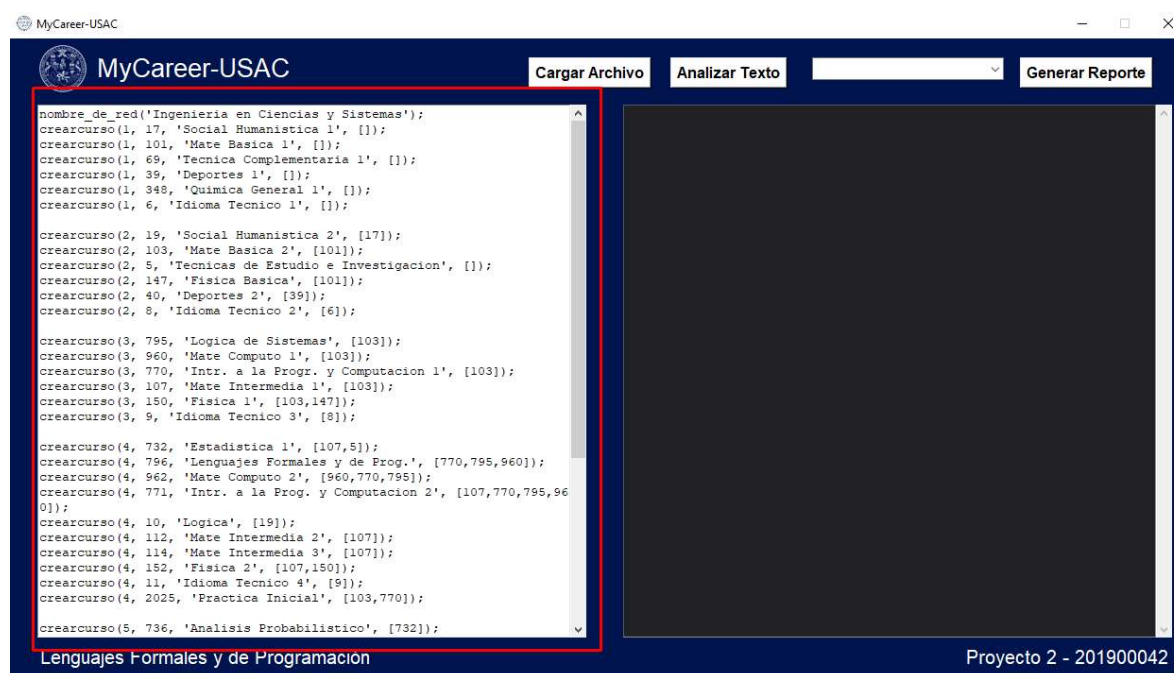
Si contiene un archivo que contenga el lenguaje lfp y quiere agregarlo al cuadro de texto entonces primero presione el botón de “Cargar Archivo”:



Luego se abrirá una pequeña ventana donde tiene que seleccionar el archivo (**tiene que ser de extensión .lfp**) :

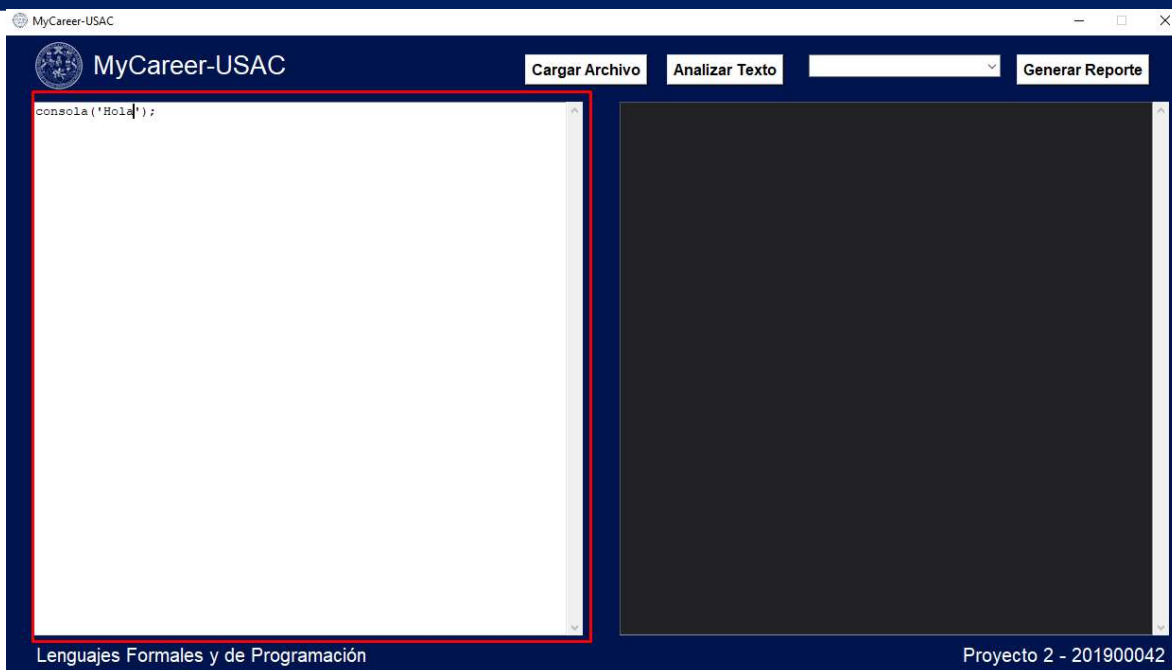


Presione “Abrir” y se cargará el archivo y aparecerá en el cuadro de texto blanco:



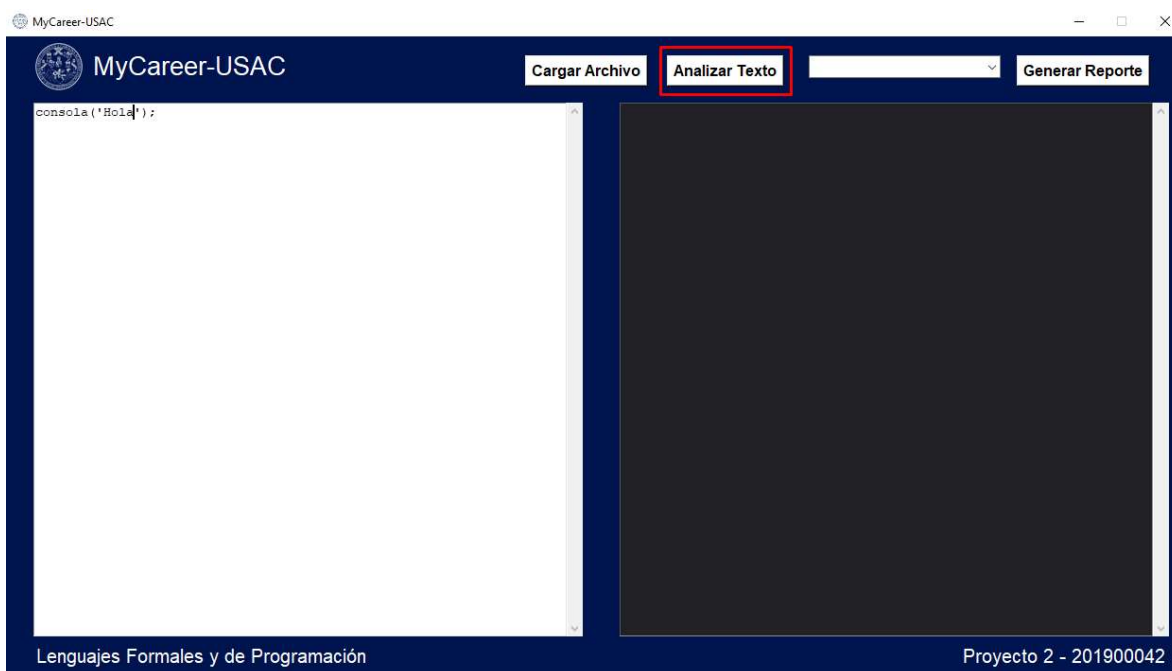
## 4. Escritura del lenguaje .lfp

Usted como usuario puede escribir en el cuadro de texto blanco el lenguaje para poder ejecutar instrucciones:



## 5. Analizar Texto

Esta opción se encuentra junto con el botón de “Cargar Archivo”:



Luego ejecutará las instrucciones escritas en el cuadro de texto blanco y se mostrarán en el cuadro de texto de color gris, y el programa ejecuta este resultado al escribir estas instrucciones:

**nombre\_de\_red:**

Al escribir esta instrucción mostrará el siguiente mensaje:

**crearcurso:**

Al escribir esta instrucción mostrará el siguiente mensaje:

**consola:**

Al escribir esta instrucción mostrara lo siguiente:

**consolaln:**

Al escribir esta instrucción mostrara lo siguiente:

**cursosPorSemestre:**

Al escribir esta instrucción mostrará lo siguiente(para que funcione bien tiene que tener los cursos creados anteriormente):

```

cursoPorSemestre(2);

Se agregó el curso Analisis Probabilistico
Se agregó el curso Org. Lenguajes y Compiladores 1
Se agregó el curso Organizacion Computacional
Se agregó el curso Estructura de Datos
Se agregó el curso Filosofia de la Ciencia
Se agregó el curso Mate Aplicada 1
Se agregó el curso Mate Aplicada 3

***** SEMESTRE: 2 *****
Código: 19
Curso: Social Humanistica 2
Requisitos:['17']

Código: 103
Curso: Mate Basica 2
Requisitos:['101']

Código: 5
Curso: Tecnicas de Estudio e Investigacion
Requisitos:[]

Código: 147
Curso: Fisica Basica
Requisitos:['101']

Código: 40
Curso: Deportes 2
Requisitos:['39']

Código: 8
Curso: Idioma Tecnico 2
Requisitos:['6']

*****

```

### cursoPorCodigo:

Al escribir esta instrucción mostrará lo siguiente(para que funcione bien tiene que tener los cursos creados anteriormente):

```

cursoPorCodigo(150);

***** SEMESTRE: 2 *****
Código: 19
Curso: Social Humanistica 2
Requisitos:['17']

Código: 103
Curso: Mate Basica 2
Requisitos:['101']

Código: 5
Curso: Tecnicas de Estudio e Investigacion
Requisitos:[]

Código: 147
Curso: Fisica Basica
Requisitos:['101']

Código: 40
Curso: Deportes 2
Requisitos:['39']

Código: 8
Curso: Idioma Tecnico 2
Requisitos:['6']

*****

*****
Curso: Fisica 1
Semestre: 3
Código: 150
Prerrequisitos: ['147', '103']
*****

```

### cursoPorNombre:

Al escribir esta instrucción mostrará lo siguiente(para que funcione bien tiene que tener los cursos creados anteriormente):

```

cursoPorNombre('Estructura de Datos');

Curso: Mate Basica 2
Requisitos:['101']

Código: 5
Curso: Tecnicas de Estudio e Investigacion
Requisitos:[]

Código: 147
Curso: Fisica Basica
Requisitos:['101']

Código: 40
Curso: Deportes 2
Requisitos:['39']

Código: 8
Curso: Idioma Tecnico 2
Requisitos:['6']

*****
Curso: Fisica 1
Semestre: 3
Código: 150
Prerrequisitos: ['147', '103']
*****
Curso: Estructura de Datos
Semestre: 5
Código: 772
Prerrequisitos: ['962', '796', '771']
*****

```

### cursosPrerrequisitos:

Al escribir esta instrucción mostrará lo siguiente(para que funcione bien tiene que tener los cursos creados anteriormente):

```

cursosPrerrequisitos(771);

Requisitos:['101']

Código: 40
Curso: Deportes 2
Requisitos:['39']

Código: 8
Curso: Idioma Tecnico 2
Requisitos:['6']

*****
Curso: Fisica 1
Semestre: 3
Código: 150
Prerrequisitos: ['147', '103']
*****
Curso: Estructura de Datos
Semestre: 5
Código: 772
Prerrequisitos: ['962', '796', '771']
*****
Curso: Intr. a la Prog. y Computacion 2
Prerrequisitos: Logica de Sistemas
Mate Computo 1
Intr. a la Progr. y Computacion 1
Mate Intermedia 1
*****

```

### cursosPostrequisitos:

Al escribir esta instrucción mostrará lo siguiente(para que funcione bien tiene que tener los cursos creados anteriormente):





## 6. Generar Reportes

Para que pueda generar Reportes de las acciones que haya hecho en el programa primero ubíquese en el cuadro de opciones y selecciona el reporte que quiere generar y seleccione el botón de “Generar Reporte”:



### Reporte de Tokens:

Esta opción es para mostrar todos los tokens generados por medio del análisis léxico del lenguaje en un archivo HTML:

Reporte de Tokens				
#	Tipo de Token	Lexema	Línea	Columna
1	tk_cursosPrerrequisitos	cursosPrerrequisitos	1	21
2	tk_parentesis	(	1	22
3	tk_entero	772	1	25
4	tk_parentesis	)	1	26
5	tk_puntoycoma	:	1	27
6	<< EOF >>	\$	2	2

### Reporte de Errores:

Esta opción es para mostrar todos los errores léxicos y sintáctico en el lenguaje escrito:



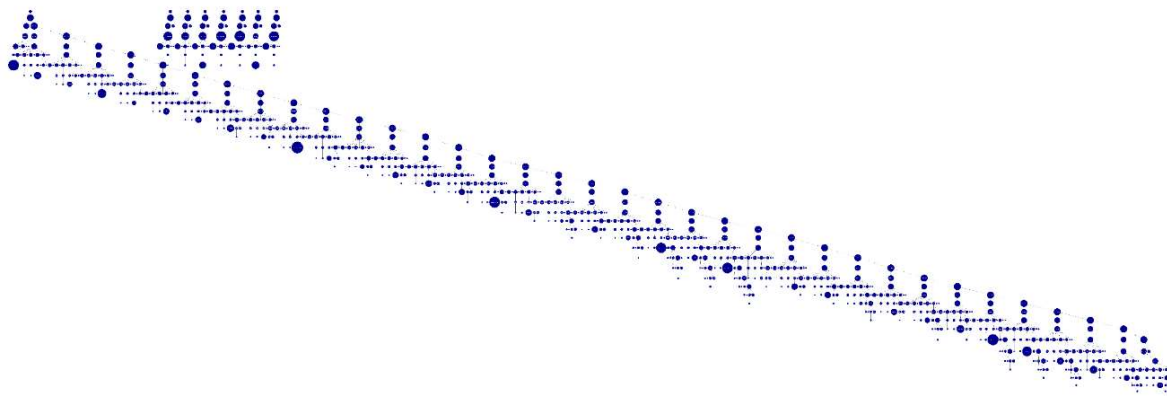
MyCareer-USAC

Reporte de Errores

#	Tipo de Error	Caracter(es)	Descripción	Línea	Columna
---	---------------	--------------	-------------	-------	---------

## Árbol de derivación:

Esta opción es para generar el grafo del árbol de derivación por medio del análisis sintáctico del lenguaje escrito:



Puede resultar muy grande por todas las instrucciones escritas y analizadas.

### **III. Recomendaciones**

- Siga las instrucciones de la estructura del lenguaje .lfp para que pueda obtener los resultados esperados de sus instrucciones.
- Tenga cuidado con los errores léxicos y sintácticos.
- Tome en cuenta que tiene que borrar el cuadro de texto blanco para que no se dupliquen las instrucciones.