



Primer Semestre 2021

Sección	Catedrático	Tutor académico
A+	Ing. Otto Amilcar Rodriguez Acosta	Javier Alberto Cabrera Puente
B+	Ing. David Estuardo Morales Ajcót	Kevin Ariel Cruz Ortíz
B-	Inga. Zulma Karina Aguirre Ordonez	Danilo Urías Coc Carlos Eduardo Hernández Molina
A-	Inga. Damaris Campos de López	Luis Manuel Morales López

Proyecto 1

Objetivos

Generales:

- Poner en práctica los conocimientos adquiridos en la clase y laboratorio.

Específicos:

- Que el estudiante utilice AFD's para la lectura de patrones.
- Que el estudiante sea capaz de identificar qué paradigmas se aplican de mejor manera para resolver problemas específicos de una manera simple y efectiva.
- Que el estudiante ponga en práctica conocimientos adquiridos sobre la lectura y manejo de archivos de texto.

Descripción

Con el fin de entender y profundizar sobre el tema de los paradigmas de programación y sobre los lenguajes de programación, específicamente Python, se le pide al estudiante que desarrolle un programa de **consola en Python** el cual debe ser capaz de leer un archivo de entrada, este archivo contiene toda la **información de un menú de restaurante**.

Se le pide que tome el archivo y a partir de éste, que genere **el menú** del restaurante en un **archivo HTML** con una interfaz agradable para el usuario. También podrá leer otro archivo el cual **contendrá una orden de pedidos** y podrá **generar la factura** correspondiente del pedido.

Las opciones que tendrá la aplicación, los detalles de los archivos a leer y la forma de interpretarlos correctamente se detalla a continuación.

Estructura de los archivos

Menú

Nombre del restaurante

El nombre del restaurante vendrá entre **comillas simples**, después de la **palabra Restaurante** y del signo igual.

```
Restaurante='NOMBRE DEL RESTAURANTE'
```

Nombre de sección

Vendrá el nombre de la sección entre **comillas simples**, seguido de **dos puntos** y en las líneas siguientes aparecerán las opciones que tiene dicha sección.

```
'NOMBRE DE SECCIÓN':
```

Opciones del menú


Las opciones del menú que tiene cada sección (**bebidas, platos principales, postres, etcétera**) vendrán **entre corchetes** uno en cada línea. Donde vendrá primero su **identificador único, nombre**, seguido del **precio en quetzales** y por último la **descripción**. Esta información vendrá separada por **puntos y comas**. El **nombre y la descripción** vendrán entre comillas simples.

```
[IDENTIFICADOR,'NOMBRE';PRECIO;'DESCRIPCIÓN']
```

Ejemplo

El siguiente ejemplo es para el **Restaurante LFP** en el cual vienen **3 secciones** en el menú. Y tiene un total de **6 opciones** para elegir.

```

restaurante='Restaurante LFP '
'Bebidas' :
[bebida_1;'Bebida #1';11.;;'Descripción Bebida 1' ]
[ bebida_2;'Bebida #2';10.50;  'Descripción Bebida 2']

'Desayunos':
[d1;'Desayuno 1';45.00;'Descripción Desayuno 1']
[d2;'Desayuno 2'; 40.  ;'Descripción Desayuno 2']
[d3;'Desayuno 3';35;'Descripción Desayuno 3']

'Postres':
[ pos_001;'Postre 1' ;25;'Descripción Postre 1']

```

Menú resultante:



Pedidos

En cada línea vendrá cada una de las opciones pedidas, primero aparecerá la cantidad pedida seguido del identificador de la opción separados por una coma.

Ejemplo

En el siguiente ejemplo se muestra la orden de 2 opciones del menú. Con una propina del 8%. Si alguno de los identificadores no se encuentra en el menú no se debe generar la factura y se debe reportar el error.

Tomar en cuenta que en la 1era línea, vienen los datos generales del cliente y el porcentaje de la propina; Nombre del cliente, NIT, dirección y Propina. Ejemplo:

```
'XX', 'xx', 'xx', 8%
2,bebida_1
4, pos_001
```

Factura resultante:

← → ↺ ↻

Restaurante LFP

Factura.html

Restarante LFP

Factura No. XX

Fecha: xx/xx/xx

Datos del Cliente

Nombre: xx

Nit: xx

Dirección: xx

Descripción			
Cantidad	Concepto	Precio	Total
2	Bebida 1	Q11.00	Q22.00
4	Postre 2	Q20.00	Q80.00
Sub total			Q102.00
Propina (8%)			Q8.16
Total			Q110.00

Generalidades

Números

Los valores numéricos deben cumplir con la siguiente expresión regular:

`[0-9]+(("[0-9"]*)?)` (**Cadenas Válidas:** 40.,14.5656,15)(**No válidas:**-40,,28,14.23.)

Cadenas

Las cadenas, la información contenida entre comillas simples, deben cumplir con la siguiente expresión regular: `'[^']'`

Identificadores

Los identificadores deben cumplir con la siguiente expresión regular:

`[a-z][a-z0-9_]*`

Tabla de tokens

Al finalizar todo el análisis, se debe de **generar un reporte de tokens** también en un archivo **HTML** con la información mínima que se muestra en el ejemplo siguiente.

No.	Lexema	Fila	Columna	Token
1	9.50	10	04	numero
2	nombre_1	2	12	identificador

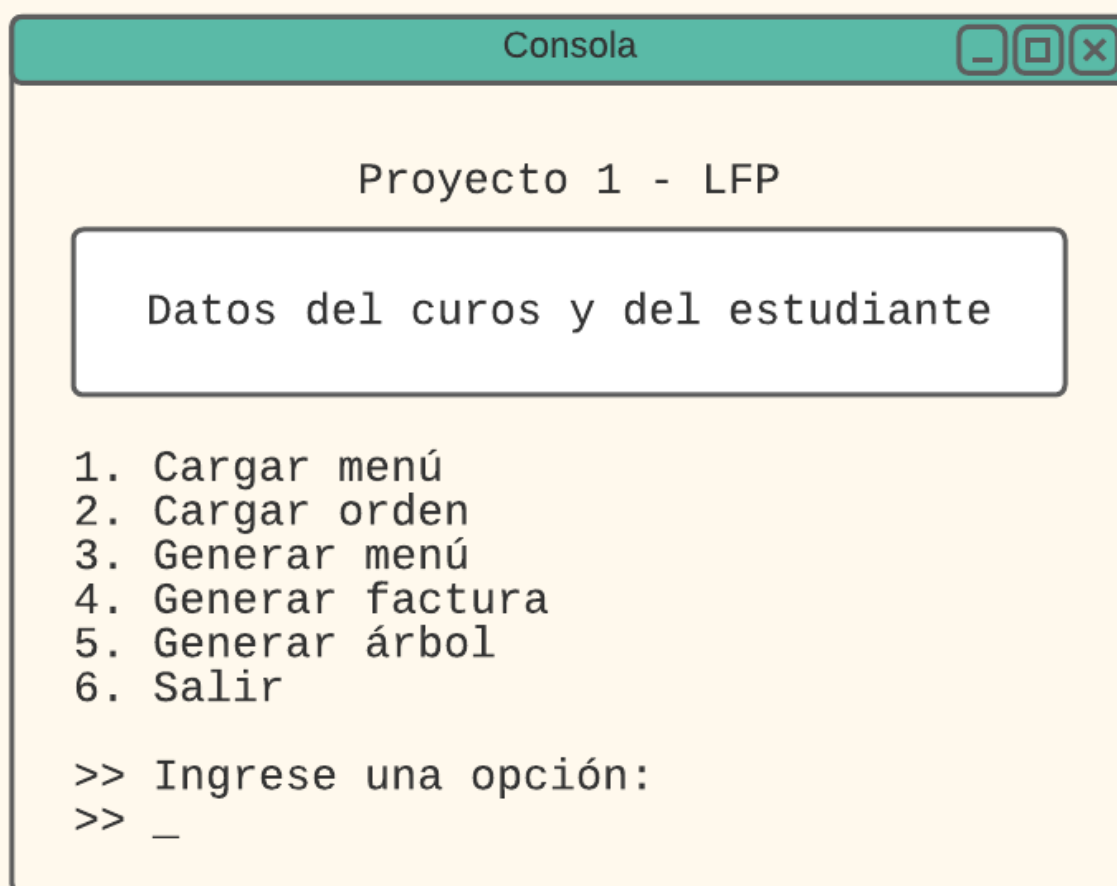
Tabla de errores

Si se llegan a encontrar **errores en los distintos archivos** de entrada **en lugar de generar el menú o factura**, se **debe generar un reporte** en HTML de los errores encontrados en el documento, con la información mínima que se muestra en el siguiente ejemplo.

No.	Fila	Columna	Carácter	Descripción
1	3	22	&	Carácter Desconocido
2	19	05	bebida--1	Identificador no valido

Características del programa

La aplicación de consola contará con una pantalla principal, en esta pantalla de inicio se debe de indicar el nombre y sección del curso, nombre y carné del estudiante y las opciones que tiene disponible el usuario, de igual manera, en esta pantalla se le debe pedir al usuario la opción a ingresar.



Opción de cargar menú

Esta opción deberá mostrar una ventana emergente que permita buscar el archivo de texto que contiene la información del menú este archivo puede estar en cualquier directorio de la computadora del usuario. Al seleccionar el archivo, este deberá ser cargado en memoria para su posterior análisis.

Opción de cargar orden

Esta opción deberá mostrar una ventana emergente que permita **buscar el archivo de texto** que contiene la información de una orden con sus respectivos pedidos, este archivo puede estar en cualquier directorio de la computadora del usuario. Al seleccionar el archivo, este deberá ser cargado en memoria para su posterior análisis.

Los archivos de entrada tendrán la extensión .lfp

Opción de generar menú

Al seleccionar esta opción se le debe preguntar al usuario **si desea poner un límite en los precios** de las distintas opciones que tiene el menú. Si se selecciona **Si**, se le debe preguntar al usuario **cual es este límite para filtrar las opciones del menú**, de seleccionar **No**, se mostrarán todas las opciones del menú.

Queda a discreción del estudiante **el diseño del menú en HTML**. Ya que este menú será el que verán los clientes tendrá una ponderación especial la presentación, debe ser creativa.

Si no se ha seleccionado **ningún archivo de menú** se deberá informar al usuario.

Opción de generar factura

Al seleccionar esta opción, **debe leer en el archivo de entrada los datos para la factura (Nombre, Nit, Dirección y % de propina)** y validar el porcentaje de la propina **(entre 0 y 100%)**.

A continuación, se leerá el archivo que contiene la orden y se **generará un archivo HTML** con la factura donde se debe mostrar el detalle con claridad. **Al subtotal se le debe agregar la propina**, y de último mostrar **el monto total**.

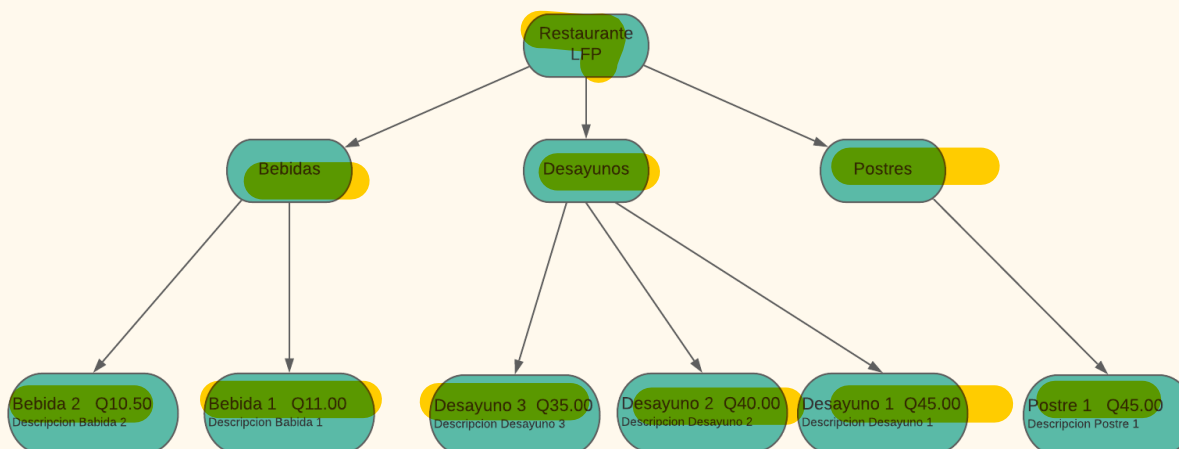
Si no se ha seleccionado **ningún archivo para la orden** se deberá informar al usuario.

Opción de generar árbol

Se deberá de utilizar **Graphviz para generar el menú ingresado en una vista de árbol** con las opciones del menú ordenadas **por el precio de menor a mayor** en cada una de las secciones.

Si no se ha seleccionado **ningún archivo de menú** se deberá informar al usuario.

Ejemplo



Se debe generar el árbol en un archivo PDF.

Salir

Al seleccionar esta opción se debe cerrar el programa.

Aclaraciones

- Se debe de utilizar Graphviz para generar la gráfica de árbol. Se recomienda consultar la documentación oficial de Graphviz <https://graphviz.org/pdf/dotguide.pdf>.
- Los valores numéricos mostrados en los menús y facturas generados deben mostrar con formato de moneda (Q y siempre 2 decimales).
- Los reportes solicitados deben ser generados en el formato indicado para que se tomen como válidos.

Entregables

- Se debe de entregar el AFD utilizado para el análisis de los distintos patrones.
- Código fuente de la aplicación funcional. En el código se deberá de poder verificar los paradigmas utilizados los cuales deberá ser capaz de explicar para comprobar la autoría de su aplicación.

Restricciones

- El proyecto debe ser implementada utilizando el lenguaje de programación Python.
- El proyecto es individual.

- Las copias de proyectos tendrán automáticamente nota de 0 puntos y se reportará a los involucrados a la Escuela de Ingeniería en Ciencias y Sistemas.
- No se recibirán proyectos después de la fecha y hora de entrega.

Entrega

- La entrega será por medio de UEDI en la actividad correspondiente.
- Tomar en cuenta que no habrá prórroga, por lo que se solicita distribuir adecuadamente su tiempo para la elaboración del proyecto.
- Para subir el proyecto deberán crear una carpeta comprimida con los entregables detallados anteriormente.
- El archivo deberá tener el siguiente formato: **Proyecto1_#CARNE_GRUPO#.zip**

Ejemplos: Proyecto1_202145678_GRUPOA, Proyecto1_202145678_GRUPOB

- **Fecha de entrega: Jueves 18 de marzo de 2021 hasta las 23:59horas No se recibirá después de la fecha y hora establecida.**