

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Laboratorio Lenguajes Formales y de Programación

Manual Técnico:

Práctica 1

Hecho por:

Walter Daniel Jiménez Hernández

201901108

Fecha: 19/08/2021

Contenido

1. Introducción
2. Objetivos
3. Conceptos previos
4. Especificación Técnica
 - a) Requisitos de Hardware
 - b) Requisitos de Software
 - i. Sistema Operativo
 - ii. Lenguaje de Programación
 - iii. Tecnologías Usadas
5. Lógica del programa
 - a) Clases Utilizadas
 - i. Alumno
 - b) Funciones utilizadas
 - i. Menu (main)
 - ii. Abrir Archivo
 - iii. Separar Datos
 - iv. Separar Alumnos
 - v. Crear Objetos
 - vi. Ordenamiento Burbuja
 - vii. Salida Consola
 - viii. Generar HTML

Introducción

El programa cumple con la función de tomar una entrada de texto con nombre y notas de alumnos de un curso en específico y mostrar esos datos ordenados o conforme a los parametros que se indican en la entrada de texto. Este manual incluye todos los aspectos técnicos que serán necesarios para poder entender la lógica del programa

Objetivo

El objetivo del programa es poder mostrar al usuario las notas ordenadas de forma ascendente o descendente como lo requiera el usuario.

- ✂ Convertir texto plano en listas de objetos
- ✂ Aplicar algoritmos de ordenamiento

Conocimientos Previos

- ✂ Programación en Python 3
- ✂ Programación Orientada a Objetos
- ✂ HTML
- ✂ Bootstrap

Especificaciones Técnicas

Requisitos de Hardware

- Procesador Pentium 2 o superior, AMD A4 o superior
- Mínimo 256 MB de Ram

Requisitos de Software

- Sistema Operativo
 - Windows 7
 - Windows 8
 - Windows 10
 - Windows 11
 - Distribuciones de Linux
- Navegador
 - Chrome
 - Edge
 - Mozilla Firefox
 - Safari
 - Opera
- Lenguaje de Programación
 - Python 3.8
- Tecnologías Usadas
 - Git
 - Visual Studio Code

Lógica del Programa

- **Clases Utilizadas**

- **Alumno**

```
1 class Alumno:
2     def __init__(self, nombre, nota):
3         self.nombre = nombre
4         self.nota = nota
```

Se crea el objeto Alumno con 2 atributos, uno es nombre y el otro es la nota

- **Funciones Utilizadas**

- **Menú (main):** Tiene un bucle while en el que se pide un número entero para elegir una opción del menú, al seleccionar una opción se genera un instrucción que sigue con el flujo del código.

```
1 if __name__ == '__main__':
2     opcion = 0
3     curso = ""
4     curso = ""
5     parametros = []
6     alumnosFinal = []
7     while(opcion != 4):
8         opcion = int(input("""
9 #####-Menú-#####
10 #
11 # Elija una opción: #
12 # 1) Cargar archivo de entrada #
13 # 2) Mostrar Reporte en Consola #
14 # 3) Exportar Reporte #
15 # 4) Salir #
16 #
17 #####
18 """))
19     if opcion == 1:
20         datos = abrirArchivo()
21         separados1 = separarDatos(datos, "\n")
22         curso = separados1[0].replace("{", "")
23         curso = curso.replace("=", "")
24         separados1.pop(0)
25         longitud = len(separados1)
26         parametro = separados1[longitud-1].replace("}", "")
27         parametro = separarDatos(parametro, ",")
28         separados1.pop(longitud-1)
29         alumnos1 = separarAlumnos(separados1)
30         alumnosFinal = crearObjetos(alumnos1)
31     elif opcion == 2:
32         html+= salidaConsola(curso, alumnosFinal, parametros, html)
33     elif opcion == 3:
34         generarHtml(html)
35     elif opcion == 4:
36         break
37     else:
38         print("\nDebe ingresar un número valido")
```

- **Abrir Archivo:** Se utiliza la herramienta TKinter para poder abrir una ventana de dialogo en el que se selecciona el archivo de entrada

```
1 def abrirArchivo():
2     Tk().withdraw()
3     archivo = askopenfilename()
4     archivo = open(archivo, "r")
5     datos = archivo.read()
6     archivo.close()
7     return datos
```

- **Separar Datos:** Separa los datos de entrada cada vez que encuentra un caracter especificado en los parametros y los agrega en una lista que es la que devuelve esta función.

```
1 def separarDatos(datos, separador):
2     temp = ""
3     listaTemp = []
4
5     for char in datos:
6         if char == separador:
7             listaTemp.append(temp.strip())
8             temp = ""
9         else:
10            temp += char
11    if temp.strip() != "":
12        listaTemp.append(temp.strip())
13    return listaTemp
```

- **Separar Alumnos:** Obtiene cada una de las filas del archivo de entrada en donde hayan alumnos, esta función quita los caracteres no deseados y devuelve una lista de alumnos con su nombre y nota.



```
1 def separarAlumnos(datos):
2     alumnos1 = []
3     for i in datos:
4         i = i.replace("<", "")
5         i = i.replace(">", "")
6         i = i.replace(",", "")
7         i = i.replace("'", "")
8         alumnos1.append(separarDatos(i, ";"))
```

- **Crear Objetos:** Crea una lista de objetos alumnos asignandole su atributo de nombre y nota.



```
1 def crearObjetos(alumnos1):
2     listafinal = []
3     for temp in alumnos1:
4         alumno = Alumno(temp[0], int(temp[1]))
5         listafinal.append(alumno)
6     return listafinal
```

- **Ordenamiento Burbuja:** Utiliza el método de ordenamiento de burbuja para ordenar de forma ascendente o descendente los datos.



```
1 def burbujaAsc(alumnos):
2     tamaño = len(alumnos)-1
3     i = 0
4     for i in range(tamaño):
5         j = 0
6         for j in range(tamaño):
7             if alumnos[j].nota > alumnos[j+1].nota:
8                 alumnos[j+1], alumnos[j] = alumnos[j], alumnos[j+1]
9
10 def burbujaDesc(alumnos):
11     i = 0
12     for i in range(len(alumnos) - 1):
13         j = 0
14         for j in range(len(alumnos) - 1):
15             if alumnos[j].nota < alumnos[j+1].nota:
16                 alumnos[j+1], alumnos[j] = alumnos[j], alumnos[j+1]
```

- **Salida Consola:** Revisa que parametros son los que se indican en la entrada para poder mostrar esos datos solicitados.
- **Generar HTML:** Genera un archivo HTML llamado "Reporte Alumnos.html" en el que se muestra toda la información que se mostro en consola de forma más amigable al usuario.

```
1 def generarHtml(html):
2     pagina = open('Reporte Alumnos.html', 'w')
3     salida = """<!DOCTYPE html>
4     <html lang="es">
5     <head>
6     <meta charset="UTF-8">
7     <meta http-equiv="X-UA-Compatible" content="IE=edge">
8     <meta name="viewport" content="width=device-width, initial-scale=1.0">
9     <link href="https://bootswatch.com/4/superhero/bootstrap.min.css" rel="stylesheet" type="text/css">
10    <title>Alumnos</title>
11    </head>
12    <body style="background: linear-gradient(to right, #141e30, #243b55);">
13    <nav class="navbar navbar-expand-lg navbar-dark bg-secondary">
14        <a class="navbar-brand" href="#">Practica LFPB+</a>
15        <a class="navbar-brand" href="#">Walter Daniel Jimenez Hernandez 201901108</a>
16    </nav>
17    """
18    salida+=html
19    salida += """</body>
20    </html>
21    """
22    pagina.write(salida)
23    pagina.close()
24    webbrowser.open_new_tab('Reporte Alumnos.html')
```