**UNIVERSIDAD DE GUADALAJARA**

**CUCEI**

**DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN**

**DEPARTAMENTO DE CIENCIAS COMPUTACIONALES**

**PROYECTO FINAL**

***EQUIPO No. 4***

*INTEGRANTES*:

**ROBLES PULIDO EFRAIN**

**RODRÍGUEZ CANO CHRISTIAN FABIAN**

**RODRÍGUEZ RICO MARIO EDUARDO**

*NOMBRE DE LA MATERIA:* **FUNDAMENTOS FILOSOFICOS DE LA COMPUTACION**

*SECCIÓN:* **D13**  *CALENDARIO:* **2021A**

*NOMBRE DEL PROFESOR:* **MARISCAL LUGO LUIS FELIPE**





**Resolver el problema**

**1.-Definición del problema**

Desarrolla un programa que muestre en pantalla el siguiente menú:

MENÚ - LISTAS

1. Llenar lista usuario (calificaciones)

2. Calcular promedio lista usuario

3. Llenar lista aleatoriamente (edades)

4. Determinar la persona de menor edad

5. Determinar la persona de mayor edad

6. Salir

Elige tu opción:

Si el usuario elige la opción 1, vamos a pedir al usuario la cantidad de calificaciones que se van a promediar y después solicitamos las respectivas calificaciones (números reales) que se guardarán en una lista, todo esto lo hará una función, dicha función regresará la lista con todos los valores proporcionados por el usuario.

Si el usuario elige la opción 2, se calculará el promedio de las calificaciones, todo esto lo llevará a cabo una función que retornará dicho promedio.

Si el usuario elige la opción 3, vamos a pedir al usuario la cantidad de edades que vamos a guardar en una lista, después se van a generar la cantidad de edades aleatoriamente (valores entre 1 y 100), todo esto lo hará una función, dicha función regresará la lista con todos los valores proporcionados por el usuario.

Si el usuario elige la opción 4, se determinará cual es la edad más pequeña, esto lo llevara a cabo una función que retornara el resultado apropiado, es decir, el menor de toda la lista.

Si el usuario elige la opción 5, se determinará cual es la edad más grande, esto lo llevara a cabo una función que retornara el resultado apropiado, es decir, el mayor de toda la lista.

**2.- Análisis del problema**

**Entrada:** Se le solicitará al usuario un dato de tipo entero que se almacenará en la variable “Opci”, permitiéndonos seleccionar una opción dentro del menú.

**Proceso:** Dentro de un ciclo while, se mostrará en pantalla un determinado menú, que dependiendo del valor de la variable “Opci” que introdujo el usuario, con la ayuda de la selectiva doble anidada se determinará qué proceso llevará.

Si se elige la opción 1 se le pedirá al usuario la cantidad de calificaciones a promediar e se ingresará las calificaciones correspondientes.

Si se elige la opción 2 se calculará el promedio de las calificaciones, con las calificaciones introducidas previamente.

Si se elige la opción 3 se le pedirá al usuario la cantidad de edades a trabajar y mediante la función aleatorio entero se generarán edades aleatorias del 1 al 100.

Si se elige la opción 4 se determinará la persona más joven entre las edades generadas previamente.

Si se elige la opción 5 se determinará la persona más adulta entre las edades generadas previamente.

Si se elige la opción 6 se saldrá del ciclo while.

**Salida:** Imprimir en pantalla el mensaje correspondiente según la opción que se eligió.

Si se eligió la opción 1, se mostrará en pantalla la lista de calificaciones que se introdujeron.

Si se eligió la opción 2, se mostrará en pantalla el promedio de las calificaciones introducidas.

Si se eligió la opción 3, se mostrará en pantalla la lista de edades aleatorias que se generaron.

Si se eligió la opción 4, se mostrará en pantalla la persona con la edad menor entre las edades.

Si se eligió la opción 5, se mostrará en pantalla la persona con la edad mayor entre las edades.

Si se eligió la opción 6, se mostrará en pantalla el mensaje de agradecimiento por utilizar el software.

**3.-Diseño del algoritmo**

**Diagramas IPO**

Plataforma: *Microsoft Word*

|  |  |  |
| --- | --- | --- |
| Diagrama IPO para la función **llenar\_lista\_usuario** | | |
| Entrada Proceso Salida | | |
| Primero se solicita que se ingrese la cantidad de calificaciones que se van a promediar, y después se ingresarán las calificaciones individualmente. | Con la cantidad de calificaciones se ajustará el tamaño de la lista list\_califs para después agregar individualmente las calificaciones deseadas. | Retorna list\_califs, la lista llena de calificaciones |

|  |  |  |
| --- | --- | --- |
| Diagrama IPO para la función **calcular\_promedio** | | |
| Entrada Proceso Salida | | |
| La lista lista\_califs\_llena es nuestro argumento para que la lista lista\_califs sea nuestro parámetro en esta funcion, en donde lista\_califs ya tendra las calificaciones almacenadas previamente y se utilizara a continuación. | Se realiza una división normal entre la suma de todos los datos de la lista lista\_califs y la cantidad de los datos de dicha lista, para obtener el promedio de las calificaciones. | Retorna la variable p, que es el promedio obtenido. |

|  |  |  |
| --- | --- | --- |
| Diagrama IPO para la función **llenar\_lista\_aleatoriamente** | | |
| Entrada Proceso Salida | | |
| Solicita al usuario que ingrese la cantidad de edades con las que se van a trabajar. | Se crea una lista con el nombre lista\_edades que será la que almacene las edades de acuerdo con la cantidad de edades deseado. Después se genera un número aleatorio del 1 al 100, para cada espacio de la lista creada previamente. | Retorna la lista lista\_edades, que tendrá los valores aleatorios (las edades). |

|  |  |  |
| --- | --- | --- |
| Diagrama IPO para la función **determinar\_menor** | | |
| Entrada Proceso Salida | | |
| La lista lista\_edad se mandará a esta función, ya tiene los valores aleatorios (las edades) a utilizar. | Con la función min y los datos almacenados en lista\_edad, se determina cual es la edad más joven y se lo asignaremos a la variable Edad\_menor. | Retorna la variable Edad\_menor. |

|  |  |  |
| --- | --- | --- |
| Diagrama IPO para la función **determinar\_mayor** | | |
| Entrada Proceso Salida | | |
| La lista lista\_edad se mandará a esta función, ya tiene los valores aleatorios (las edades) a utilizar. | Con la función max y los datos almacenados en lista\_edad, se determina cual es la edad más adulta y se lo asignaremos a la variable Edad\_mayor. | Retorna la variable Edad\_mayor. |

**Pseudocódigo**

#Robles Pulido Efrain

#Rodríguez Cano Christian Fabian.

#Rodríguez Rico Mario Eduardo.

#seccion D13

#calendario 2021A

#Algoritmo donde utilizaremos un menú para que el usuario elija una opción que necesite trabajar

Importar aleatorio

#Funcion que realizaran las llamadas del menú

definición principal ():

Opci 0

Mientras Opci <> 6: #Ciclo en donde se selecciona una opcion o el 6 para salir

Imprimir ("MENÚ - LISTAS")

Imprimir ("1. Llenar lista usuario (calificaciones)")

Imprimir ("2. Calcular promedio lista usuario")

Imprimir ("3. Llenar lista aleatoriamente (edades)")

Imprimir ("4. Determinar la persona de menor edad")

Imprimir ("5. Determinar la persona de mayor edad")

Imprimir ("6. Salir")

Entrada ("Elige tu opcion: ") # Selección de la opcion

Leer (Opci)

Si Opci = 1: #Llena la lista con calificaciones

lista\_califs\_llenallenar\_lista\_usuario ()

Imprimir ("Las calificaciones son = ", lista\_califs\_llena) #Lista llena con los valores introducidos

Imprimir (" ") #Salto de línea

Sino:

Si Opci = 2: #Se calcula el promedio de las anteriores calificaciones

Imprimir ("El promedio es = ", format(calcular\_promedio(lista\_califs\_llena), \

'.2f')) #promedio de los valores ya introducidos

Imprimir (" ") #Salto de línea

Sino:

Si Opci = 3: #Se llena la lista de edades aleatoriamente con la función

lista\_edadllenar\_lista\_aleatoriamente ()

Imprimir ("Las edades son = ", lista\_edad)

Imprimir (" ") #Salto de línea

Sino:

Si Opci = 4: #Se determinara la edad más joven

Imprimir ("La persona más joven tiene ", determinar\_menor(lista\_edad), "años")

Imprimir (" ") #Salto de linea

Sino:

Si Opci = 5: #Se determinara la edad más adulta

Imprimir ("La persona más adulta tiene", determinar\_mayor(lista\_edad), "años")

Imprimir (" ") #Salto de linea

Sino:

Si Opci = 6: #Salida del programa

Imprimir ("Gracias por utilizar este software")

#Funcion que permite llenar una lista con calificaciones

definición llenar\_lista\_usuario ():

Entrada ("¿Cuantas calificaciones se van a promediar?: ")

Leer (cant\_calif)

list\_califs [0.0] \* cant\_calif #Se crea una lista inicializada con la misma cantidad

#de materias para agregar las calificaciones deseadas

index0

Mientras index < cant\_calif:

Entrada ("Dame una calificacion: ") #Se introduce las calificaciones

Leer (list\_califs[index])

index+=1

retornar list\_califs #Regresamos la lista llena de calificaciones

#Funcion para calcular el promedio de las calificaciones de la lista

definición calcular\_promedio (lista\_califs):

indiceacum p 0

Mientras indice < longitud (lista\_califs): #longitud (lista\_califs) nos dara la longitud de la lista llena

Acum acum+lista\_califs[indice]

Índice += 1 #indice= indice+1

p acum / indice #Division normal para obtener el promedio

retornar p #Se regresa el promedio

#Funcion para generar una lista aleatoria de edades

definición llenar\_lista\_aleatoriamente ():

Entrada ("¿Cuantas edades vamos a trabajar?: ")

Leer (cant\_ed)

lista\_edades [0] \* cant\_ed #Se crea una lista inicializada con la misma

index0 #cantidad de edades a utilizar

Mientras index < cant\_ed:

lista\_edades[index] aleatorio.aleatorioentero(1,100)#Se generara un num. aleatorio del 1 al 100

index+=1

retornar lista\_edades #Se regresa la lista llena de edades

#Funcion para encontrar la edad más joven de la lista de edades, es decir, el mínimo

definición determinar\_menor (lista\_edad):

Edad\_menorminimo (lista\_edad)

retornar Edad\_menor #Se regresa la edad más joven

#Funcion para encontrar la edad más adulta de la lista de edades, es decir, el máximo

definición determinar\_mayor (lista\_edad):

Edad\_mayor maximo (lista\_edad)

retornar Edad\_mayor #Se regresa la edad más adulta

#LLamada a la función principal

Principal ()

**4.-Codificación**

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Imagen que contiene Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

**5.- Conclusiones personales**

**Robles Pulido Efraín*:***

Fue una actividad muy larga ya que implementamos los conocimientos que se vieron a lo largo del curso, me ayudo también a estudiar para el examen ya que tuve que repasar el funcionamiento de las funciones utilizadas como el *len, max,min, random y etc,*  y el control selectivo doble anidado y el ciclo while que nos ayudaron a manipular y guardar los valores en las listas para este proyecto. Además, gracias a las funciones pudimos dividir las tareas a realizar para tener un orden al realizar el código. En esta materia puede ampliar mis conocimientos de programación en PYTHON, ya que solía programar en Arduino y en PICs lo cual tiene muchas funciones, pero en esta materia pude conocer como funciona el sistema y la programación en PYTHON.

**Rodríguez Cano Christian Fabian:**

Para concluir con este proyecto cabe decir que aquí utilizamos mucho de lo aprendido en clases anteriores, fue un trabajo un poco largo o confuso de realizar por los temas nuevos que vimos un poco rápido, pero este proyecto lo fuimos realizando con el grupo y el profesor durante las sesiones de clases, lo cual ayudo a entender mejor entre todos el cómo funciona y como lo íbamos estructurando juntos, como se mencionó vimos temas nuevos como las listas, el min y el max. También temas algo ya repasados en prácticas como el uso de la selectiva doble. Fue un proyecto en el cual se trabajó lo aprendido en clases por lo que lo considero bueno y de ayuda para entender de los futuros trabajos, temas y proyectos que nos esperan más adelante.

**Rodríguez Rico Mario Eduardo:**

En este proyecto utilizamos todo lo aprendido en nuestro curso de Fundamentos Filosóficos de la Computación. Se pudo crear un menú con 6 diferentes programas utilizando las listas, las funciones “min” y “max”, los índices, bucles, ciclos while, etc. Me agrado que también se puedan ingresar códigos diferentes con la importación de librerías, en este caso se utilizó la importación de la librería “random” para poder crear en nuestro menú las edades aleatorias. Sin duda fue muy interesante el crear este proyecto junto a mis compañeros a pesar de las dificultades que se pudieron presentar tanto en los diagramas como en el código fuente. Le agradezco a cada uno de ellos por apoyarme a poder realizar este proyecto y las demás prácticas. Al igual que le agradezco al profesor por impartir el curso y mostrarnos lo bonito que es Python.