UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA  
FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN  
LICENCIATURA EN INGENIERÍA EN SISTEMAS DE INFORMACIÓN  
CURSO: - ALGORITMOS  
CATEDRÁTICO: Ing. Miguel Catalán



Proyecto funcional completo y borrador de manual técnico y de usuario

Alumna : jheimy Alejandra paredez ac

# BORRADOR DE MANUAL TECNICO

INTROCCION

El programa esta basado en el lenguaje de programación Python no cuenta con interfaz, se utiliza por medio de la consola de Python, cuenta con 7 opciones cada una de ellas actuara con forme su nombre indica, para seleccionar una opción debe de ingresar el numero de la opción que desee y presionar enter para comenzar a ejecutar la acción, a continuación, se mostrará y se explicara cada opción.

Opción 1: agregar curso

Esta opción le permite ingresar el nombre del curso que desee, además en este curso se almacenara todas las notas que ingresara, puede ingresar más de un curso, pero debe de hacerlo de 1 en 1 no mas de 1 a la vez. Ejemplo

“Agregar Curso”

1

Matemáticas

Curso matemáticas agregado

Opción 2: eliminar curso

Esta opción le proporciona la decisión de eliminar algún curso que no desee en el sistema de ser asi presione 2 y enter, para ejecutar la acción, ingrese el nombre del curso que quiera eliminar, Importante poner el nombre correctamente como se muestra en la lista de cursos que tenga ya en el sistema de no ser asi la acción no se ejecutara, ya escrito el nombre del curso que eliminara presione enter para eliminar el curso.

Lista de cursos: Matemáticas

“eliminar curso “

2

Matemáticas

Curso eliminado de la lista: Matemáticas

Opción 3: ingresar notas y calcular el promedio

Esta opción tendrá insertar el numero de notas que ingresara al sistema, el numero de notas deber ser 1 en adelante, presione 3, ingrese la el número de notas que ingresará, ingrese el valor de las notas, la petición de ingrese el valor de notas aparecerá de acuerdo al numero de notas que ingreso previamente, al finalizar se calculara el promedio automáticamente.

Ingrese el numero de notas

3

Ingrese el valor de la nota 1:

2

Ingrese el valor de la nota 2:

4

Ingrese el valor de la nota 3:

5

El promedio de las notas es de: 3 .67

Opción 4: ordenar curso

Esta opción funcionara solamente si tiene cursos agregados, al tener mas de 1 curso en la lista de cursos puede usar esta opción para ordenarlos, se ordenarán en orden alfabético empezando por la A luego a, B, b etc. Presione el numero 4 y enter para comenzar la acción.

Lista de cursos: Algebra, precálculo, contabilidad, algoritmos, Matemáticas

Ordenar cursos

4

Lista de cursos: Algebra, algoritmos, contabilidad, Matemáticas

Opción 5: ver Historial

Esta opción le permitirá ver la acción realizas anteriormente, se mostrará todas las opciones seleccionadas anteriormente, si no tienen acciones anteriores no le mostrara nada, presione el 5 para ejecutar la opción.

Ver historial

5

Agregar curso

Eliminar curso

Ordenar curso

Notas y promedio

Opción 6: eliminar la última operación del historial

Esta opción permite borrar la última opción realizada del historial, esto no afectara en el resultado que realizo dicha opción, presione 6 para ejecutar la acción.

Eliminar la última operación del historial

6

Ultima operación borrada

Opción 7: salir

Si desea salir del programa presione el numero 7 y enter, no se guardan datos ni acciones

Salir

7

Salió del programa

Arquitectura del sistema de notas por materia

El presente programa está diseñado bajo una **arquitectura modular y funcional**, lo que significa que las tareas del sistema se dividen en **módulos independientes** representados por funciones específicas. Cada función tiene una **responsabilidad bien definida**, facilitando la legibilidad, el mantenimiento y la ampliación futura del código.

En primer lugar, se declara una **lista global llamada materias**, que almacena los nombres de las asignaturas disponibles. Esta lista sirve como punto central de referencia para las demás funciones, ya que permite validar entradas del usuario y organizar los cursos.

La función menu() actúa como **interfaz principal con el usuario**, mostrando mensajes de bienvenida e instrucciones iniciales. Esta función no recibe parámetros ni retorna valores, su función principal es guiar la interacción del usuario antes de realizar acciones específicas.

La función ingreso\_de\_nota() representa el **núcleo funcional del sistema**, permitiendo seleccionar una materia, ingresar un número variable de notas y calcular el promedio correspondiente. Mediante el uso de estructuras como try-except, el programa controla errores de entrada (por ejemplo, cuando el usuario escribe un texto en lugar de un número), lo que mejora la **robustez y estabilidad del sistema**.

El uso de listas dinámicas (Notas=[]) permite almacenar múltiples calificaciones sin conocer de antemano cuántas ingresará el usuario. Además, se utiliza un ciclo for junto a un while True interno para validar cada calificación individualmente, asegurando que los datos sean siempre numéricos y válidos.

El programa incluye un **ciclo principal while True dentro de la función main()**, que mantiene el menú en ejecución continua hasta que el usuario decida salir. Esta estructura refleja el patrón clásico de **bucle de control principal** utilizado en programas interactivos.

La función ordenar\_materias() aplica el concepto de **ordenamiento e interacción con estructuras globales**, mostrando cómo organizar las materias de forma alfabética mediante un algoritmo de inserción (aunque su implementación puede ampliarse). Esto fomenta el aprendizaje de algoritmos de ordenamiento básicos en Python.

Por otro lado, la función eliminar\_ultima\_operacion() introduce la idea de manejar un **historial de acciones del usuario**, aplicando métodos de listas como .pop() para eliminar la última operación registrada. Este enfoque demuestra la posibilidad de extender el sistema hacia un modelo más completo con registro de operaciones.

Finalmente, el bloque if \_\_name\_\_ == "\_\_main\_\_": asegura que el programa solo se ejecute automáticamente cuando es ejecutado directamente, no cuando se importa como módulo en otro script. Este detalle sigue las **buenas prácticas de programación en Python**.

En conjunto, esta arquitectura combina **estructuras de control, validación de datos, manejo de errores y modularidad**, ofreciendo una base sólida para un sistema educativo interactivo. Además, el código está pensado para ser escalable: se pueden agregar nuevas funciones como exportar promedios, guardar datos en archivos o integrar interfaces gráficas en el futuro.