



Profundiza más

Recurso de Profundización

Clase 13: Arreglos y funciones (Parte 1)

Autor: Damián Nicolalde Rodríguez

Objetivo del Documento:

El objetivo de este recurso de profundización es integrar los conceptos aprendidos en la clase 13 sobre funciones, parámetros, retorno de valores y alcance de variables. Utilizaremos estos conocimientos junto con el uso de arreglos unidimensionales y multidimensionales (matrices) en Python para resolver un problema práctico de programación.

Descripción del Ejercicio:

Vamos a crear un sistema de gestión de calificaciones para un curso. Este sistema permitirá a los profesores ingresar las calificaciones de los estudiantes, calcular sus promedios, clasificarlos según su rendimiento, y almacenar los resultados en una lista (arreglo unidimensional). Además, se utilizarán matrices para almacenar las calificaciones de manera más estructurada.

El sistema también implementará varias funciones que descomponen el problema en tareas más pequeñas y manejables, utilizando los conceptos de parámetros, retorno de resultados, y alcance de variables.

Requisitos

1. Crear una función para ingresar las calificaciones de los estudiantes.
2. Calcular el promedio de cada estudiante usando una función que reciba las calificaciones como parámetro.
3. Clasificar a los estudiantes como "Aprobado", "Reprobado", o "Excelente" según su promedio.
4. Almacenar las calificaciones y resultados en una matriz bidimensional.
5. Permitir la modificación de las calificaciones usando funciones que alteren la matriz global de calificaciones.

Contenido del Recurso de Profundización

1. Definir Funciones para la Gestión de Estudiantes

Vamos a crear varias funciones que permitan al sistema realizar distintas tareas: agregar calificaciones, calcular promedios, clasificar estudiantes, y mostrar los resultados.



Profundiza más

2. Uso de Arreglos y Funciones para Organizar los Datos

Usaremos arreglos unidimensionales para almacenar las calificaciones de los estudiantes y una matriz bidimensional para representar las calificaciones de todos los estudiantes en diferentes exámenes.

3. Alcance de Variables

Trabajaremos con variables locales y globales en diferentes funciones para manejar las calificaciones y los resultados de los estudiantes.

Código

```
# Definir una lista vacía global para almacenar los estudiantes y sus calificaciones
```

```
estudiantes = []
```

```
# Función para ingresar las calificaciones de un estudiante
```

```
def ingresar_calificaciones(nombre):
```

```
    # Lista de calificaciones del estudiante
```

```
    calificaciones = []
```

```
    # Ingresar las calificaciones para cada examen (por ejemplo, 3 exámenes)
```

```
    for i in range(3): # Suponemos que hay 3 exámenes
```

```
        calificacion = float(input(f'Ingrese la calificación para el examen {i+1} de {nombre}: '))
```

```
        calificaciones.append(calificacion)
```

```
    # Agregar el estudiante y sus calificaciones a la lista global de estudiantes
```

```
    estudiantes.append([nombre, calificaciones])
```

```
# Función para calcular el promedio de un estudiante
```

```
def calcular_promedio(calificaciones):
```

```
    return sum(calificaciones) / len(calificaciones)
```

```
# Función para clasificar a los estudiantes según su promedio
```

```
def clasificar_estudiantes():
```

```
    # Usamos una matriz que contiene las calificaciones de los estudiantes
```

```
    for estudiante in estudiantes:
```

```
        nombre = estudiante[0]
```



Profundiza más

```
calificaciones = estudiante[1]
promedio = calcular_promedio(calificaciones)

# Clasificar a los estudiantes según el promedio
if promedio >= 9:
    estado = "Excelente"
elif promedio >= 6:
    estado = "Aprobado"
else:
    estado = "Reprobado"

# Mostrar el resultado
print(f'Estudiante: {nombre} | Promedio: {promedio:.2f} | Estado:
{estado}')

# Función para mostrar el listado de estudiantes y sus calificaciones
def mostrar_estudiantes():
    for estudiante in estudiantes:
        nombre = estudiante[0]
        calificaciones = estudiante[1]
        print(f'{nombre}: Calificaciones: {calificaciones}')

# Función para modificar las calificaciones de un estudiante
def modificar_calificaciones(nombre):
    for estudiante in estudiantes:
        if estudiante[0] == nombre:
            # Modificar las calificaciones de ese estudiante
            for i in range(len(estudiante[1])):
                nueva_calificacion = float(input(f'Ingrese la nueva calificación
para el examen {i+1} de {nombre}: '))
                estudiante[1][i] = nueva_calificacion
            print(f'Las calificaciones de {nombre} han sido modificadas.')
            return
    print(f'Estudiante {nombre} no encontrado.')

# Función principal del sistema de gestión
def sistema_gestion():
```



Profundiza más

```
while True:
    print("\n1. Ingresar calificaciones de un estudiante")
    print("2. Calcular y clasificar estudiantes")
    print("3. Mostrar todos los estudiantes y sus calificaciones")
    print("4. Modificar calificaciones de un estudiante")
    print("5. Salir")

    opcion = input("Seleccione una opción: ")

    if opcion == "1":
        nombre = input("Ingrese el nombre del estudiante: ")
        ingresar_calificaciones(nombre)

    elif opcion == "2":
        clasificar_estudiantes()

    elif opcion == "3":
        mostrar_estudiantes()

    elif opcion == "4":
        nombre = input("Ingrese el nombre del estudiante a modificar: ")
        modificar_calificaciones(nombre)

    elif opcion == "5":
        print("Gracias por usar el sistema.")
        break
    else:
        print("Opción no válida. Intente nuevamente.")

# Ejecutar el sistema de gestión
sistema_gestion()
```

Explicación Detallada

1. Definición de Funciones:

- `ingresar_calificaciones`: Esta función solicita al usuario que ingrese las calificaciones de un estudiante para varios exámenes y las guarda en una lista. Esta lista luego se agrega a la lista global `estudiantes`.



Profundiza más

- `calcular_promedio`: Calcula el promedio de las calificaciones de un estudiante sumando las calificaciones y dividiendo por la cantidad de exámenes.
 - `clasificar_estudiantes`: Utiliza la función `calcular_promedio` para obtener el promedio de cada estudiante y clasifica al estudiante según su promedio (Excelente, Aprobado, Reprobado).
 - `mostrar_estudiantes`: Imprime todos los estudiantes con sus calificaciones.
 - `modificar_calificaciones`: Permite al usuario modificar las calificaciones de un estudiante ya ingresado en la lista global.
2. **Estructura de Datos:**
- **Arreglo Unidimensional:** En este caso, la lista de calificaciones de un solo estudiante es un arreglo unidimensional (es decir, una lista de valores numéricos).
 - **Matriz (Arreglo Multidimensional):** La lista estudiantes es una lista de listas, donde cada sublista contiene el nombre de un estudiante y su lista de calificaciones.
3. **Alcance de las Variables:**
- Las variables `nombre` y `calificaciones` son locales dentro de las funciones donde se utilizan.
 - La lista `estudiantes` es global y se puede modificar desde cualquier parte del programa.
4. **Métodos Comunes en Listas:**
- `append()`: Usado para agregar nuevos estudiantes y sus calificaciones.
 - `sum()`: Usado en la función `calcular_promedio` para obtener la suma de las calificaciones.