

Tarea 5: Examen corto, calculo de IMC

Alan Andrés Mérida Morales, 202100023^{1,*}

¹Facultad de Ingeniería, Escuela de Ingeniería Mecánica Eléctrica,
Universidad de San Carlos, Ciudad Universitaria, Zona 12, Guatemala.

I. OBJETIVOS

- Desarrollar un programa en Octave que permita calcular el índice de masa corporal, categorizarlo según su valor y ofrecer las opciones de guardar, leer y borrar la información en un archivo de texto, todo ello sin utilizar funciones de funcionamiento.

II. INTRODUCCIÓN

Este proyecto en Octave está diseñado para proporcionar una herramienta interactiva que permite calcular el Índice de Masa Corporal (IMC) de un usuario, clasificarlo en categorías específicas y gestionar la información a través de un archivo de texto. El programa ofrece un menú sencillo con opciones para calcular el IMC, visualizar los datos almacenados, eliminar el archivo de registro y salir del programa. La solución emplea un enfoque de manejo de errores robusto para asegurar que el usuario reciba mensajes claros en caso de fallos. Este proyecto demuestra cómo combinar cálculos básicos con operaciones de archivo en Octave, utilizando `disp` para mostrar la información de manera efectiva.

III. RESULTADOS

A. Códigos realizados

```
1
2 if(exist('OCTAVE_VERSION', 'builtin')==0)
3 % estamos en octave
4 pkg load signal;
5 end
6
7 bajoPeso = 'Bajo peso';
8 pesoNormal = 'Peso normal';
9 sobrePeso = 'Sobre peso';
10 %menu principal
11 % Inicialización
12 continuar = true;
13
14 opcion = 0;
15 while opcion ~= 4
16 % Menú de opciones
17 disp('1. Calcular IMC');
18 disp('2. Leer información del archivo');
19 disp('3. Borrar información del archivo');
20 disp('4. Salir');
21 opcion = input('Ingrese su elección: ');
22
23 switch opcion
24 case 1
25 % Calcular IMC
26 try
27     nombre = input('Ingrese su nombre: ', 's');
28     peso = input('Ingrese su peso en kg: ');
29     altura = input('Ingrese su altura en metros: ');
30
31     imc = peso / (altura^2);
32     disp(['Su IMC es: ', num2str(imc)]);
33
34 % Categorizar IMC
35 if imc < 18.5
36     categoria = bajoPeso;
37 elseif imc < 24.9
38     categoria = pesoNormal;
39 elseif imc < 31.9
40     categoria = sobrePeso;
41 else
42     categoria = 'Obesidad';
43 end
```

Figura 1: Código realizado para calcular el IMC de un usuario

```
44 disp(['Categoría: ', categoria]);
45
46 % Guardar en archivo de texto 'imc.txt'
47 archivo = fopen('imc.txt', 'a'); % Abre el archivo en modo append
48 fprintf(archivo, '%s\n', '%s\t', categoria, '%s\n', nombre, imc,
49 categoria);
50 fclose(archivo);
51 disp('Información guardada correctamente.');
```

Figura 2: Código realizado para calcular el IMC de un usuario

```
81
82 case 4
83 % Salir
84 disp('Gracias por usar el programa');
85
86 otherwise
87     disp('Opción no válida.');
```

Figura 3: Código realizado para calcular el IMC de un usuario

* 3690273450101@ingenieria.usac.edu.gt

B. Explicación del código

El código comienza verificando si se está ejecutando en Octave y carga el paquete `signal` si es necesario. Luego, presenta un menú principal al usuario con opciones para calcular el Índice de Masa Corporal (IMC), leer información del archivo, borrar el archivo de datos, y salir del programa. El menú se presenta en un bucle `while` que continúa ejecutándose hasta que el usuario selecciona la opción de salir.

Cuando el usuario selecciona la opción para calcular el IMC, el programa solicita el nombre, peso y altura. Con estos datos, calcula el IMC usando la fórmula:

$$IMC = \frac{peso}{altura^2} \quad (1)$$

determina la categoría del IMC (Bajo peso, Normal, Sobrepeso u Obesidad) basada en el valor calculado. La información del IMC y la categoría se muestran utilizando `disp`, y los datos se guardan en un archivo de texto `imc.txt` para su posterior referencia.

i el usuario elige leer el archivo, el programa intenta abrir `imc.txt` y muestra su contenido línea por línea usando `disp`. Si el archivo no se puede abrir, se notifica al usuario. Para borrar el archivo, el programa elimina `imc.txt` y confirma la acción con un mensaje. Finalmente, si el usuario selecciona la opción de salir, el programa termina el bucle y finaliza. En cada caso, el código utiliza manejo de errores con `try-catch` para asegurar que cualquier problema se maneje adecuadamente, proporcionando mensajes de error claros cuando sea necesario.

C. Repositorio privado

Repositorio creado en Github

IV. CONCLUSIONES

- * El programa demuestra la capacidad de Octave para manejar entradas del usuario y realizar cálculos básicos de manera interactiva. La implementación permite calcular el Índice de Masa Corporal (IMC), clasificarlo según las categorías establecidas y gestionar la información de forma eficiente mediante la lectura y escritura en archivos de texto.
- * La funcionalidad de guardar, leer y borrar datos en un archivo de texto proporciona una forma práctica de almacenar y gestionar información de manera persistente. Además, el uso de bloques `try-catch` asegura que el programa maneje errores de manera adecuada, previniendo que fallos en la entrada de datos o en el manejo de archivos interrumpen el flujo del programa.
- * La elección de utilizar `disp` para mostrar información y resultados simplifica el código y hace que la salida sea fácilmente legible para el usuario. Aunque no se emplea formateo avanzado de texto, el enfoque elegido es suficiente para un uso básico y efectivo en la demostración de funcionalidades.
- * Este proyecto puede servir como base para futuras ampliaciones, como la integración de gráficos para visualizar datos o la incorporación de funcionalidades más avanzadas como análisis de tendencias a partir de múltiples entradas. También se pueden implementar medidas adicionales de validación de datos para mejorar la robustez del programa.