## Tarea 5: Examen corto, calculo de IMC

Alan Andrés Mérida Morales, 202100023<sup>1,\*</sup>

<sup>1</sup>Facultad 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 corportal, categorizarlo según su valor y ofrecer las opciones de guardar, leer y borrar la informacion en un archivo de texto, todo ello sin utilizar funcionesfuncionamiento.

### 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 Affacial ('CONNEL UMERION', 'builtin') ==0)
1 Setimose an octave
pkp load signal;
end

2 bajopace 'Bajo paso';
pasolornale 'Pean onnal';
pasolornale 'Rea onnal';
pasolornale 'Teap on 'Pean on 'Pe
```

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

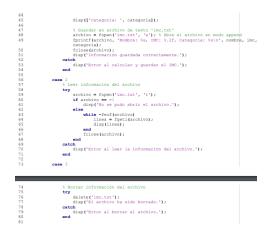


Figura 2: Código realizado para calcular el IMC de un usuario



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

<sup>\* 3690273450101@</sup>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:

$$IMC = \frac{peso}{altura^2} \tag{1}$$

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 interrumpan 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.