



CALCULO BMI

Métricas de Rendimiento

Daniel Alejandro Olarte Ávila

Universidad Sergio Arboleda

Universidad Sergio Arboleda Cl. 74 #14-14

Bogotá, Colombia

Correo: danielolarte2012@gmail.com

Escuela de Ciencias Exactas e Ingeniería

Profesor: John Corredor

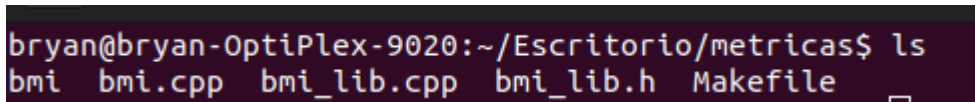
03/08/2022

ENUNCIADO:

Realizar un programa en Linux a través de Consola para poder calcular el BMI de una persona y poder imprimir el resultado a través de categorías para dejar claro como esta dicha persona respecto del peso y la altura.

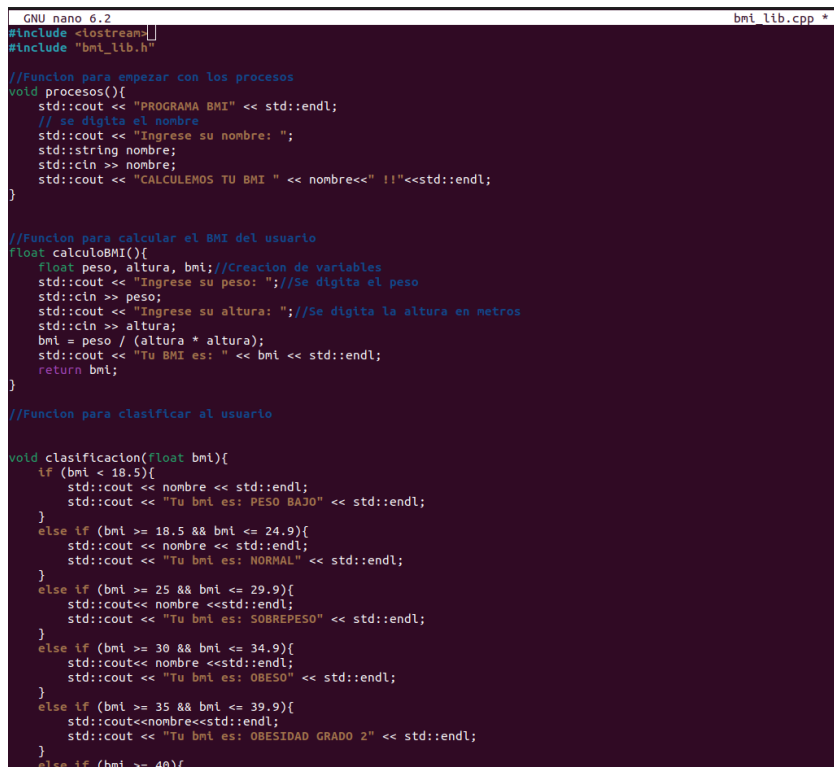
CÓDIGO:

- **LS:** La siguiente imagen se refiere al usar el comando "ls" en la consola en la ruta dentro de la carpeta donde se esta trabajando el proyecto, para evidenciar los 4 archivos creados en clase para que funcione el programa de calcular el bmi, y además se puede evidenciar el .exe bmi el cual hace referencia al ejecutable



```
bryan@bryan-OptiPlex-9020:~/Escritorio/metricas$ ls
bmi  bmi.cpp  bmi_lib.cpp  bmi_lib.h  Makefile
```

- **Bmi_lib.cpp:** En este archivo .cpp se encontrará el código donde se muestra que hará cada función, los mensajes para el usuario, las peticiones para poder trabajar con los datos obtenidos, la clasificación según el resultado, etc.



```
GNU nano 6.2                                bmi_lib.cpp *
#include <iostream>
#include "bmi_lib.h"

//Funcion para empezar con los procesos
void procesos(){
    std::cout << "PROGRAMA BMI" << std::endl;
    // se digita el nombre
    std::cout << "Ingrese su nombre: ";
    std::string nombre;
    std::cin >> nombre;
    std::cout << "CALCULEMOS TU BMI " << nombre << " !!"<<std::endl;
}

//Funcion para calcular el BMI del usuario
float calculoBMI(){
    float peso, altura, bmi; //Creacion de variables
    std::cout << "Ingrese su peso: "; //Se digita el peso
    std::cin >> peso;
    std::cout << "Ingrese su altura: "; //Se digita la altura en metros
    std::cin >> altura;
    bmi = peso / (altura * altura);
    std::cout << "Tu BMI es: " << bmi << std::endl;
    return bmi;
}

//Funcion para clasificar al usuario
void clasificacion(float bmi){
    if (bmi < 18.5){
        std::cout << nombre << std::endl;
        std::cout << "Tu bmi es: PESO BAJO" << std::endl;
    }
    else if (bmi >= 18.5 && bmi <= 24.9){
        std::cout << nombre << std::endl;
        std::cout << "Tu bmi es: NORMAL" << std::endl;
    }
    else if (bmi >= 25 && bmi <= 29.9){
        std::cout << nombre << std::endl;
        std::cout << "Tu bmi es: SOBREPESO" << std::endl;
    }
    else if (bmi >= 30 && bmi <= 34.9){
        std::cout << nombre << std::endl;
        std::cout << "Tu bmi es: OBESO" << std::endl;
    }
    else if (bmi >= 35 && bmi <= 39.9){
        std::cout << nombre << std::endl;
        std::cout << "Tu bmi es: OBESIDAD GRADO 2" << std::endl;
    }
    else if (bmi >= 40){
```

- **Bmi.cpp:** En este archivo encontramos el método principal, el cual llamada cada función para realizar los procedimientos y poder calcular el BMI solicitado por el usuario

```
GNU nano 6.2 bmi.cpp *  
  
#include <stdio.h>  
#include "bmi_lib.h"  
  
int main()  
{  
    procesos();  
    clasificacion(calculoBMI());  
    return 0;  
}
```

- **Bmi_lib.h:** El archivo .h se refiere al header o archivo de cabecera del programa, este representa las buenas practicas al separar los métodos en un archivo aparte, este nos servirá en el caso de un programa mas extenso, al dejar los métodos que se usan mas frecuentes en un archivo .h y así realizar un modo económico para ahorrar tiempo al programar o modificar el código del código en estos archivos

```
GNU nano 6.2 bmi_lib.h *  
  
#ifndef BMI_LIB_H_INCLUDED  
#define BMI_LIB_H_INCLUDED  
#include "bmi_lib.cpp"  
#include <iostream>  
  
//Funcion de bienvenida  
void procesos();  
  
//Funcion de calculo BMI  
float calculoBMI();  
  
//Funcion para la clasificacion del BMI  
  
void clasificacion(float bmi);  
  
#endif
```

- **Makefile**

```

GNU nano 6.2                                     Makefile
gcc = g++
FLAGS= -ansi -pedantic -Wall
CFLAGS = -lm
PROGS = bmi

bmi:
    $(gcc) $(FLAGS) -o bmi bmi.cpp

clean:
    $(rm) $(PROGS)*.o

```

SOLUCIÓN DEL PROBLEMA DE CLASE:

En clase evidenciamos un error, al momento de compilar el código, la solución que le di para solucionarlo fue mas sencilla que la que presento mi compañero en clase, en mi caso solo puse una línea de código que falto, `#include "bmi_lib.cpp"` en el archivo header, esto ocurría porque no le habíamos conectado en el header algún archivo .cpp para que pudiera empezar a enviar sus métodos y poder compilar para calcular el bmi del usuario.

```

GNU nano 6.2                                     bmi_lib.h *
#ifndef BMI_LIB_H_INCLUDED
#define BMI_LIB_H_INCLUDED
#include "bmi_lib.cpp"
#include <iostream>

//Funcion de bienvenida
void procesos();

//Funcion de calculo BMI
float calculoBMI();

//Funcion para la clasificacion del BMI

void clasificacion(float bmi);

#endif

```

```
#include "bmi_lib.cpp"
```

COMPILAR EL PROYECTO:

Para compilar el proyecto se abre la terminal y se crea el ejecutable en el caso que no se tenga ya, esto con el comando `g++` (nombre del archivo `cpp`) `-o` (nombre del ejecutable).

Ejemplo:

```
g++ bmi.cpp -o bmi
```

Una vez se tenga el ejecutable se compilara el proyecto con el comando

```
./ (nombre del ejecutable)
```

En nuestro ejemplo `./bmi`

```
daniel@daniel-VirtualBox:~$ cd Escritorio/  
daniel@daniel-VirtualBox:~/Escritorio$ cd BMI/  
daniel@daniel-VirtualBox:~/Escritorio/BMI$ ./bmi  
PROGRAMA BMI  
Ingrese su nombre:  
█
```

SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/BMI$ ./bmi  
PROGRAMA BMI  
Ingrese su nombre:  
daniel  
CALCULEMOS TU BMI daniel !!  
Ingrese su peso:  
65  
Ingrese su altura:  
1.70  
Tu BMI es: 22.4913  
daniel  
Tu bmi es: NORMAL  
daniel@daniel-VirtualBox:~/Escritorio/BMI$ █
```