



Taller ejercicios C++

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

10/08/2022

TALLER EJERCICIOS METRICAS DE RENDIMIENTO

RESUMEN:

En este taller se realizarán 9 ejercicios de C++, cada uno con un problema diferente, donde se podrá probar diferentes habilidades donde cada vez va aumentando la dificultad a la hora de programar y así poder demostrar y aprender el conocimiento adquirido en este lenguaje, se realizara los ejercicios por terminal en Linux, se mostrara como se crean con comandos los archivos, como se modifican y su respectiva salida.

INTRODUCCIÓN:

Se realizará el taller en el lenguaje de programación c++, el cual es un lenguaje fuertemente tipado, y se lleva muy bien con la programación orientada a objetos, en el taller se realizara desde ejercicios básicos hasta unos más complejos, abarcando así diferentes opciones que se puede realizar en el lenguaje para cumplir el objetivo de cada programa

PRIMER PASO: Abrir la terminal para crear cada ejercicio del taller

```
daniel@daniel-VirtualBox: $
```

SEGUNDO PASO: Nos direccionamos a la ruta en donde vamos a crear
los proyectos

```
daniel@daniel-VirtualBox:~$ cd Escritorio/  
daniel@daniel-VirtualBox:~/Escritorio$ cd Trabajos\ U/  
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U$ cd Metricas/  
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas$ cd PrimerTallerEjercicios/  
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

CREACION DE LOS PROYECTOS:

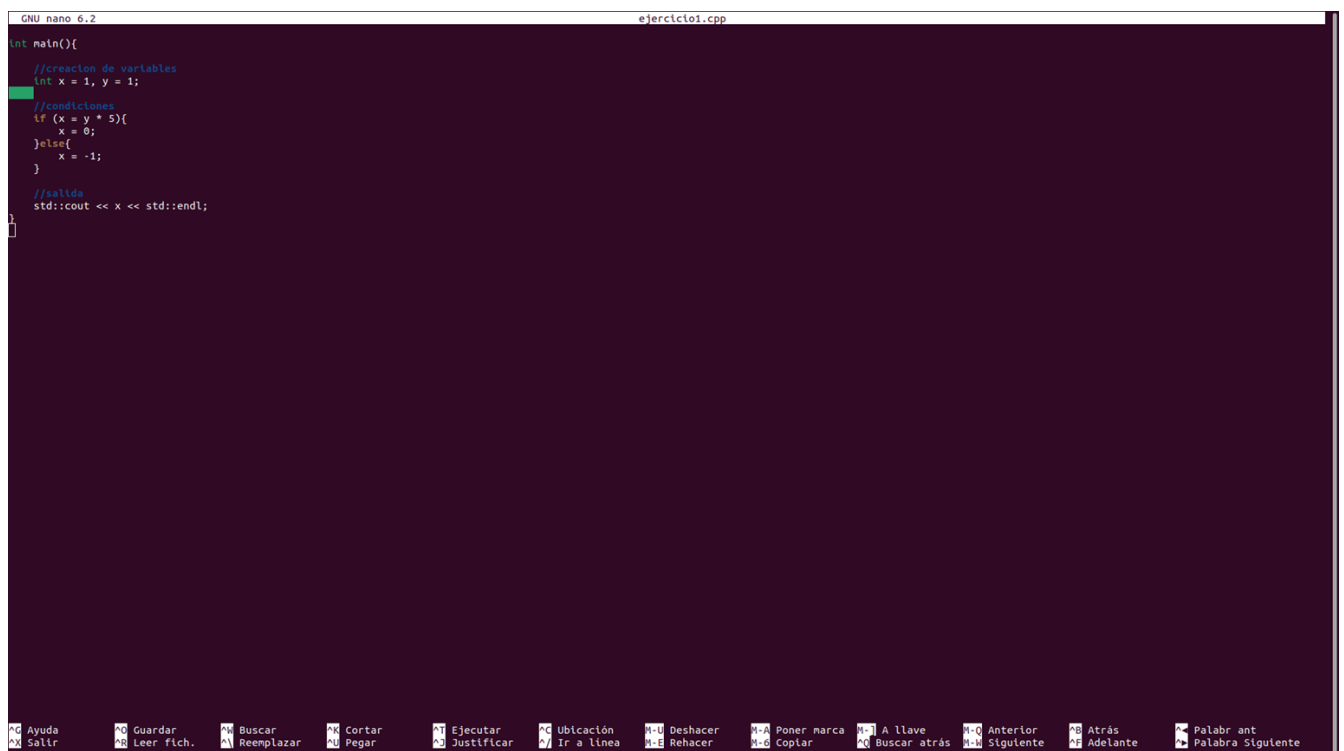
Primer Ejercicio:

CREACIÓN DEL PROYECTO: Se escribe el comando “nano Ejercicio1.cpp” para crear el
archivo de c++ y empezar a escribir el código

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio1.cpp
```



CÓDIGO:



CREACIÓN DEL EJECUTABLE: Se crea el ejecutable con el comando “g++ (nombre del archivo) -o (nombre del ejecutable)”

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio1.cpp -o ejercicio1
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ls
documento.odt  ejercicio1  ejercicio1.cpp  ejercicio2.cpp  ejercicio3.cpp  ejercicio4.cpp  ejercicio5.cpp  ejercicio6.cpp  ejercicio7.cpp  ejercicio8.cpp  ejercicio9.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Se ejecuta el comando “./(nombre del ejecutable)” para compilar el proyecto

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ls
documento.odt  ejercicio1  ejercicio1.cpp  ejercicio2.cpp  ejercicio3.cpp  ejercicio4.cpp  ejercicio5.cpp  ejercicio6.cpp  ejercicio7.cpp  ejercicio8.cpp  ejercicio9.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio1
```

SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio1
0
```

Segundo Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio2.cpp
```

CÓDIGO:

```
GNU nano 6.2
/**
 * @file ejercicio2.cpp
 * @author Daniel Olarte
 * @version 0.1
 * @date 2022-08-10
 *
 * @copyright Copyright (c) 2022
 */

#include <iostream>

int main(){

    //creacion de variables
    int x = 1, y = 1;

    //condiciones
    if (x == 1){
        if (y == 0){
            x = 10;
        }
    }else{
        x = -1;
    }

    //salida
    std::cout << x << std::endl;
}
```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio2.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio2.cpp -o ejercicio2
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio2
1
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Tercer Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio3.cpp
```

CÓDIGO:

```
GNU nano 6.2
/**
 * @file ejercicio3.cpp
 * @author Daniel Olarte
 * @version 0.1
 * @date 2022-08-10
 *
 * @copyright Copyright (c) 2022
 */

#include <iostream>

int main(){

    //creacion de variables
    int x = 0;

    //condiciones
    for (x = 'a'; x <= 'z'; x += 10){
        std::cout << x << std::endl;
    }
}
```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio3.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio3.cpp -o ejercicio3
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio3
97
107
117
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Cuarto Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio4.cpp
```

CÓDIGO:


```
GNU nano 6.2
/**
 * @file ejercicio4.cpp
 * @author Daniel Olarte
 * @version 0.1
 * @date 2022-08-10
 *
 * @copyright Copyright (c) 2022
 */

#include <iostream>

int main(){

    //creacion de variables
    int x = 0, y = 0;

    //condiciones
    for (x = 6; x > 0; x -= 2){
        for (y = 0; y < 2; y++){
            std::cout << x-y << std::endl;
        }
    }
}
```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio4.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio4.cpp -o ejercicio4
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio4
6
5
4
3
2
1
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Quinto Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio5.cpp
```

CÓDIGO:

```
GNU nano 6.2
/**
 * @file ejercicio5.cpp
 * @author Daniel Olarte
 * @version 0.1
 * @date 2022-08-10
 *
 * @copyright Copyright (c) 2022
 *
 */

#include <iostream>

//metodo para imprimir la variable x
void fnx(int x){
    if (x){ //si x es distinto de 0
        std::cout << x << std::endl;
    }
}

int main() {
    //creacion de variables
    int i, a = 1234;
    //condiciones
    for (i = 0; i < 4; i++){
        fnx(a = a/10);
    }
}
```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio5.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio5.cpp -o ejercicio5
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio5
123
12
1
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Sexto Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio6.cpp
```

CÓDIGO:

```
/*Realizar un programa que calcule e imprima la suma de los múltiplos de 5
comprendidos entre dos valores a y b. El programa no permitirá introducir valores
negativos para a y b, y verificará que a es menor que b. Si a es mayor que b,
intercambiará estos valores.*/

/**
 * @file ejercicio6.cpp
 * @author Daniel Olarte
 * @brief Múltiplos de 5 en un rango (Ejercicio 6)
 * @version 0.1
 * @date 2022-08-10
 *
 *
 * @copyright Copyright (c) 2022
 *
 */

#include <iostream>

int main(){
    // Declaración de variables
    int a, b, suma = 0;

    // Entrada de datos
    std::cout << "Introduce un valor para a: ";
    std::cin >> a;
    std::cout << "Introduce un valor para b: ";
    std::cin >> b;

    //condicional que no acepte valores negativos para a y b
    if (a < 0 || b < 0){
        std::cout << "No se aceptan valores negativos" << std::endl;
    }else{
        //condicional que verifique que a es menor que b
        if (a > b){
            int aux = a;
            a = b;
            b = aux;
        }
        //ciclo que recorre los valores de a y b
        for (int i = a; i <= b; i++){
            if (i % 5 == 0){ //condicional que verifique que el valor de i sea múltiplo de 5
                suma += i;
            }
        }
    }
}
```

5 50 140000

```
    //imprime los resultados
    std::cout<<"\n\nRESULTADOS"<<std::endl;
    std::cout << "\nLa suma de los múltiplos de 5 de los numeros entre " << a << " y " << b << " es: " << suma << "\n\n" << std::endl;
}

return 0;
}
```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio6.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio6.cpp -o ejercicio6
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio6
Introduce un valor para a: 6
Introduce un valor para b: 32

RESULTADOS

La suma de los múltiplos de 5 de los numeros entre 6 y 32 es: 100

daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Séptimo Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio7.cpp
```

CÓDIGO:

```
GNU nano 6.2                                                                 ejercicio7.cpp
/*Si quiere averiguar su número de Tarot, sume los números de su fecha de nacimiento y a continuación redúzcalos a un único dígito; por ejemplo si su fecha de nacimiento fuera 17 de octubre de 1992, los cálculos a realizar serían:

17 + 10 + 1992 = 2019 => 2 + 0 + 1 + 9 = 12 => 1 + 2 = 3
lo que quiere decir que su número de Tarot es el 3.*/

/**
 * @file ejercicio7.cpp
 * @author Daniel Olarte
 * @brief Numero del Tarot (Ejercicio 7)
 * @version 0.1
 * @date 2022-08-10
 *
 * @copyright Copyright (c) 2022
 */

#include <iostream>

int main(){
    // Declaración de variables
    int dia, mes, anio, suma = 0;

    // Entrada de datos
    std::cout << "Introduce un valor para dia: ";
    std::cin >> dia;
    std::cout << "Introduce un valor para mes: ";
    std::cin >> mes;
    std::cout << "Introduce un valor para anio: ";
    std::cin >> anio;

    //condicional que no acepte valores negativos para dia y mes y anio y que verifique que mes y dia esten entre 1 y 12
    if (dia < 0 || mes < 0 || anio < 0 || dia > 31 || mes > 12){
        std::cout << "No se aceptan valores negativos" << std::endl;
    }else{

        //suma los valores de dia, mes y anio
        suma = dia + mes + anio;

        //ciclo que suma los digitos de la suma
        while (suma > 9){
```

[71 líneas leídas]

```
        while (suma > 9){
            suma = suma/10 + suma%10;
        }

        //imprime los resultados
        std::cout<<"\n\nRESULTADOS"<<std::endl;
        std::cout << "\nEl numero de Tarot de " << dia << "/" << mes << "/" << anio << " es: " << suma << "\n\n" << std::endl;
    }
    return 0;
}
```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio7.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio7.cpp -o ejercicio7
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio7
Introduce un valor para día: 10
Introduce un valor para mes: 04
Introduce un valor para año: 2001

RESULTADOS

El numero de Tarot de 10/4/2001 es: 8

daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$
```

Octavo Ejercicio:

CREACIÓN DEL PROYECTO:

```
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio8.cpp
```

CÓDIGO:

/*

Realizar un programa que genere la siguiente secuencia de dígitos:

```

          1
        2 3 2
      3 4 5 4 3
    4 5 6 7 6 5 4
  5 6 7 8 9 8 7 6 5
6 7 8 9 0 1 0 9 8 7 6
7 8 9 0 1 2 3 2 1 0 9 8 7
8 9 0 1 2 3 4 5 4 3 2 1 0 9 8

```

• El número de filas estará comprendido entre 11 y 20 y el resultado aparecerá centrado en la pantalla como se indica en la figura.

*/

/**

* @file ejercicio8.cpp

* @author Daniel Olarte

* @brief Programa Piramide con secuencia (Ejercicio 8)

* @version 0.1

* @date 2022-08-10

*

* @copyright Copyright (c) 2022

*

*/

#include <iostream>

int main(){

//creacion de variables

int cantF;

//entrada de datos

std::cout << "Digite la cantidad de filas: " << std::endl;

std::cin >> cantF;

//condiciones

for (int x = 1; x <= cantF; x++){

for (int k=10 ; k>=x; k--){


```

for (int k=10; k>=x; k--){
    std::cout << " ";
}

if(x==1){
    std::cout << x << std::endl;
}else{
    int z = ((x*2))-1;

    for (int y = 0; y <= ((x*2)-2); y++){
        if((x+y) >= x*2){
            z-=1;
            if(z>9){
                std::cout << z%10 << " ";
            }else{
                std::cout << z << " ";
            }
        }else{
            if(x+y >9){
                std::cout << (x+y)%10 << " ";
            }else{
                std::cout << x+y << " ";
            }
        }
    }
    std::cout << std::endl;
}
}
}
}

```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```

daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio8.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio8.cpp -o ejercicio8
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio8
Digite la cantidad de filas:
11
      1
    2 3 2
  3 4 5 4 3
4 5 6 7 6 5 4
5 6 7 8 9 8 7 6 5
6 7 8 9 0 1 0 9 8 7 6
7 8 9 0 1 2 3 2 1 0 9 8 7
8 9 0 1 2 3 4 5 4 3 2 1 0 9 8
9 0 1 2 3 4 5 6 7 6 5 4 3 2 1 0 9
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0
1 2 3 4 5 6 7 8 9 0 1 0 9 8 7 6 5 4 3 2 1
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ 

```

Noveno Ejercicio:

CREACIÓN DEL PROYECTO:

```

daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio9.cpp

```

CÓDIGO:

```
/*
Realizar un programa que calcule el importe a pagar por un vehículo al circular
por una autopista. El vehículo puede ser una bicicleta, una moto, un coche o un
camión. Para definir el conjunto vehículos utilizaremos un tipo enumerado (vea en
el capítulo 2 los tipos enumerados). El importe se calculará según los siguientes
datos:
```

- Un importe fijo de 100 unidades para las bicicletas.
- Las motos y los coches pagarán 30 unidades por Km.
- Los camiones pagarán 30 unidades por Km más 25 unidades por Tm.

La presentación en pantalla de la solución, será de la forma siguiente:

```
1 - Bicicleta
2 - Moto
3 - Coche
4 - Camión
5 - Salir
Seleccione la opción deseada: 4
Kilómetros y toneladas? 100 50
Importe = 4250
*/
```

```
/**
 * @file ejercicio9.cpp
 * @author Daniel Olarte
 * @brief Programa Vehiculos en autopista (Ejercicio 9)
 * @version 0.1
 * @date 2022-08-10
 *
 *
 * @copyright Copyright (c) 2022
 *
 */
```

```
#include <iostream>
```

```
void camionOp(){
    int importe, km, tn;
    //system("cls");
    std::cout<<"\n\t\t\tCAMION\n\n"<<std::endl;

    std::cout<<"\tDigite los km recorridos: ";
    std::cin>>km;
```

```
std::cout<<"\n\tDigite las Tn del camion: ";
std::cin>>tn;

importe = (km*30) + (tn * 25);

std::cout<<"\n\tIMPORTE: "<<importe;
//system("pause");
//system("cls");
}

void cocheOp(){
    int importe, km;
    //system("cls");
    std::cout<<"\n\t\t\tCOCHE\n\n"<<std::endl;

    std::cout<<"\tDigite los km recorridos: ";
    std::cin>>km;

    importe = km*30;

    std::cout<<"\n\tIMPORTE: "<<importe;
    //system("pause");
    //system("cls");
}

void motoOp(){
    int importe, km;
    //system("cls");
    std::cout<<"\n\t\t\tMOTO\n\n"<<std::endl;

    std::cout<<"\tDigite los km recorridos: ";
    std::cin>>km;

    importe = km*30;

    std::cout<<"\n\tIMPORTE: "<<importe;
    //system("pause");
    //system("cls");
}

void bicicletasOp(){
    int importe = 100;
    //system("cls");
    std::cout<<"\n\t\t\tBICICLETA\n\n"<<std::endl;
```

```

GNU nano 6.2
    std::cout<<"\n\t\t\tBICICLETA\n\n"<<std::endl;
    std::cout<<"\n\t\tIMPORTE: "<<importe;
    //system("pause");
    //system("cls");
}

void menu(){
    int men;
    std::cout<<"\t\t\tMENU:\n"<<std::endl;
    std::cout<<"\t1) Bicicleta"<<std::endl;
    std::cout<<"\t2) Moto"<<std::endl;
    std::cout<<"\t3) Coche"<<std::endl;
    std::cout<<"\t4) Camion"<<std::endl;
    std::cout<<"\t0) Salir\n"<<std::endl;
    std::cout<<"\tSeleccione la opcion deseada: ";
    std::cin>>men;

    switch(men){
        case 1:
            bicicletasOp();
            break;
        case 2:
            motoOp();
            break;
        case 3:
            cocheOp();
            break;
        case 4:
            camionOp();
            break;
        case 0:
            exit(0);
            break;
        default:
            std::cout<<"\n\n\t\t\tDIGITO UN VALOR INCORRECTO, VUELVA A INTENTAR\n\n";
            break;
    }
    menu();
}

```

```

int main(){
    menu();
    //system("pause");
}

```

CREACIÓN DEL EJECUTABLE Y SALIDA:

```

daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ nano ejercicio9.cpp
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ g++ ejercicio9.cpp -o ejercicio9
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ./ejercicio9
MENU:

1) Bicicleta
2) Moto
3) Coche
4) Camion
0) Salir

Seleccione la opcion deseada: 1

BICICLETA

IMPORTE: 100          MENU:

1) Bicicleta
2) Moto
3) Coche
4) Camion
0) Salir

Seleccione la opcion deseada: 4

CAMION

Digite los km recorridos: 24

Digite las Tn del camion: 11

IMPORTE: 995          MENU:

1) Bicicleta
2) Moto
3) Coche
4) Camion
0) Salir

Seleccione la opcion deseada: 0
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ █

```

VISTA DE ARCHIVOS CREADOS:

```

daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ ls
documento.odt  ejercicio1.cpp  ejercicio2.cpp  ejercicio3.cpp  ejercicio4.cpp  ejercicio5.cpp  ejercicio6.cpp  ejercicio7.cpp  ejercicio8.cpp  ejercicio9.cpp
ejercicio1    ejercicio2      ejercicio3      ejercicio4      ejercicio5      ejercicio6      ejercicio7      ejercicio8      ejercicio9
daniel@daniel-VirtualBox:~/Escritorio/Trabajos U/Metricas/PrimerTallerEjercicios$ █

```

CONCLUSION:

Al realizar los ejercicios pudimos demostrar las habilidades con el lenguaje, desde operadores aritméticos, ciclos, métodos, hasta menús para tomar diferentes decisiones diferentes y que se pueda repetir el código, a través de esto pudimos evidenciar ejercicios que nos hicieron pensar y ser creativos para llegar a que el programa cumpla su función, y mostrar su respectiva salida, además de esto perfeccionamos la habilidad con los comandos de Linux para realizar cada uno de los ejercicios, y crear sus ejecutables para así poder compilar exitosamente cada uno de los ejercicios.