

Profesor: John Jairo Corredor Franco

Estudiante: Oscar Julián Reyes Torres – 1052410442

Implementación

Captura de Pantalla

1) ¿Cuál es el resultado del siguiente programa?

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Primer Punto Taller
6  */
7
8  #include <iostream>
9
10 using namespace std;
11
12 main() {
13
14     int x = 1, y = 1;
15     if(x = y * 5)
16         x=0;
17     else
18         x = -1;
19
20     printf("%d\n", x);
21 }
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distrib

0

Process exited after 0.5464 seconds with return value 0
Presione una tecla para continuar . . .

Resultados de la compilación (Log) Depuración Ver resultados

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2.
- Output Size: 1,83441543579102 MiB
- Compilation Time: 2,45s

Como se puede evidenciar, el resultado es 0 dado que dentro de la **if** se le asignó el valor de **x = y * 5**, lo cual generó que x fuera igual a 5 pero una vez dentro del condicional igualo a **x = 0** siendo este el resultado a imprimir.

2) ¿Cuál es el resultado del siguiente programa?

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Segundo Punto Taller
6  */
7
8  #include <iostream>
9
10 using namespace std;
11
12 main() {
13
14     int x = 1, y = 1;
15     if(x == 1)
16         if(y==0)
17             x=10;
18     else
19         x = -1;
20
21     printf("%d\n", x);
22 }
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Di

-1

Process exited after 0.4975 seconds with return value 0
Presione una tecla para continuar . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad
- Output Size: 1,83441543579102 MiB
- Compilation Time: 2,55s

El resultado como se puede evidenciar es 1, dado que en el primer condicional sí cumple con que la variable entera **x** sea 1 pero en el segundo condicional **y** no es igual a 0, por tanto, sale del condicional **if** y entra al **else** donde a **x** le asigna un valor de -1 siendo este el que se imprime

3) ¿Cuál es el resultado del siguiente programa?

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Tercer Punto Taller
6  */
7
8  #include <iostream>
9
10 using namespace std;
11
12 main() {
13
14     int x = 0;
15     for (x = 'a'; x <= 'z'; x +=10){
16         printf ("%c ", x);
17     }
18 }
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida

```
a k u
-----
Process exited after 0.5462 seconds with return value 0
Presione una tecla para continuar . . .
```

Compilation results...

```
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida
- Output Size: 1,83441543579102 MiB
- Compilation Time: 2,67s
```

El resultado corresponde a los valores del código **ASCII** al iterar sobre la variable **x**, inicializada con la letra '**a**' cuyo ASCII es 97 y debe incrementar de 10 en 10, hasta que sea menor igual a z (en código ASCII es 122); por tanto, las que cumplen con esta condición son los caracteres **a, k, u** correspondientes a los valores 97, 107 y 117

4) ¿Cuál es el resultado del siguiente programa?

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Cuarto Punto Taller
6  */
7
8  #include <iostream>
9
10 using namespace std;
11
12 main() {
13
14     int x = 0, y = 0;
15     for (x = 6; x > 0; x -= 2)
16         for (y = 0; y < 2; y++)
17             printf("%d ", x-y);
18
19 }
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida

```
6 5 4 3 2 1
-----
Process exited after 0.6157 seconds with return value 0
Presione una tecla para continuar . . .
```

Compilation results...

```
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida
- Output Size: 1,83441543579102 MiB
- Compilation Time: 3,05s
```

Como podemos observar, el programa asigna el valor de 6 a la variable **x**, en el primer ciclo **for** y en el segundo **for** empieza a iterar aumentando la variable **y**, es decir, primero itera el bucle interior y va restando a la variable **x=6**, después itera el bucle externo el cual va disminuyendo el valor de **x** siempre y cuando sea mayor a 0. Los valores mostrados son los resultados de la resta de **x,y**.

5) ¿Cuál es el resultado del siguiente programa?

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Quinto Punto Taller
6  */
7
8  #include <iostream>
9  using namespace std;
10
11 void fnx(int x){
12     if (x) printf("%d ", x);
13 }
14
15 main() {
16
17     int i, a = 1234;
18     for (i = 0; i < 4; i++)
19         fnx(a = a/10);
20
21 }
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribu

123 12 1

Process exited after 0.6213 seconds with return value 0
Presione una tecla para continuar . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2
- Output Size: 1,83443546295166 MiB
- Compilation Time: 2,78s

Como se puede evidenciar, en el programa se crea una función de tipo **void** el cual como parámetro necesita recibir número entero el cual imprime, en la función principal o **main** se declaran dos variables enteras, '**i**' que funcionará de índice y una variable '**a**' con un valor de 1234, haciendo uso de un ciclo **for** se hace el llamado a la función declarada anteriormente, llamada "**fnx**" 3 veces enviándole como parámetro la variable "**a**" divida entre 10, para la primera iteración **a=1234/10 → a=123** y de esta forma se imprimen los valores respectivos de 123, 12 y 1.

6) 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.

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Sexto Punto Taller
6  */
7
8  #include <iostream>
9  using namespace std;
10
11 /*
12 Función sumar los múltiplos de 5 entre un número a
13 y un número b
14 */
15
16 int sumMultiplos(int a, int b){
17     int i=0, sum=0;
18     for(i = a; i <= b; i++){
19         if(i%5 == 0){
20             sum += i;
21         }
22     }
23
24     return sum;
25 }
26
27 /*
28 Se requiere implementar un programa que sume los
29 números múltiplos de 5 de dos números mayores a 0
30 */
31
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida\Computacion_Paralela_y_Distribuida\1er...

----- Calculadora de la suma de los multiplos de 5 entre dos numeros -----

Ingrese el primer numero mayor a 0

3

Ingrese el segundo numero mayor a 0

45

Suma de los multiplos de 5 entre 3 y 45 es igual a: 225

Process exited after 5.1 seconds with return value 0

Presione una tecla para continuar . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela
- Output Size: 1,83314609527588 MiB
- Compilation Time: 2,75s

```

32 int main(int argc, char **argv) {
33     cout << "\t ----- Calculadora de la suma de los multiplos de 5 entre dos numeros ----- \n" << "\n";
34     int num1 = 0, num2 = 0, sum = 0;
35
36     do{
37         cout << "Ingrese el primer numero mayor a 0" << endl;
38         cin >> num1;
39     }while(num1 < 0);
40
41     do{
42         cout << "Ingrese el segundo numero mayor a 0" << endl;
43         cin >> num2;
44     }while(num2 < 0);
45
46     if(num1 < num2){
47         sum = sumMultiplos(num1, num2);
48         cout << "\n Suma de los multiplos de 5 entre " << num1 << " y " << num2 << " es igual a: " << sum << endl;
49     }else{
50         sum = sumMultiplos(num2, num1);
51         cout << "\n Suma de los multiplos de 5 entre " << num2 << " y " << num1 << " es igual a: " << sum << endl;
52     }
53 }
54

```

Como se puede evidenciar el programa cumple el requerimiento solicitado, dado que satisface la suma de los múltiplos de 5 presentes dentro de un número **a** y un número **b**, también verifica por medio de un ciclo **while** que los números ingresados no sean negativos o menores a 0 y en dado caso que el número **a** sea mayor al número **b**, se intercambien.

7) Realizar un programa que permita evaluar la serie:

$$\sum_{a=0}^b \frac{1}{x + ay}$$

```

1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Septimo Punto Taller
6  */
7
8
9  #include <iostream>
10 using namespace std;
11
12 // Creación de una función que evalúe serie
13
14 double serialEvaluation(int b, double x, double y){
15     int a;
16     double serial = 0;
17     for(a = 0; a <= b; a++){
18         serial += 1/(x+a*y);
19     }
20     return serial;
21 }
22
23
24 //Implementación de una aplicación que evalúe la serie dada
25 int main(int argc, char** argv) {
26     int b = 0;
27     double x = 0, y = 0, serial = 0;
28
29     cout << "\t ----- Calculador de Series ----- \n" << endl;
30
31     do{
32         cout << "Digite el valor (mayor que cero ) hasta donde desea realizar la sumatoria (limite superior)" << endl;
33         cin >> b;
34     }while(b < 0);
35
36     do{
37         cout << "Digite el valor de x (mayor que cero)" << endl;
38         cin >> x;
39     }while(x < 0);
40
41     do{
42         cout << "Digite el valor de y (mayor que cero)" << endl;
43         cin >> y;
44     }while(y < 0);
45
46     serial = serialEvaluation(b, x, y);
47     cout << "La serie hasta el limite superior " << b << " es igual a: " << serial << endl;
48
49 }

```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida\Computacion_Paralela_y_Distribuida

```

----- Calculador de Series -----
Digite el valor (mayor que cero ) hasta donde desea realizar la sumatoria (limite superior)
50
Digite el valor de x (mayor que cero)
40
Digite el valor de y (mayor que cero)
5
La serie hasta el limite superior 50 es igual a: 0.410679


-----
Process exited after 9.123 seconds with return value 0
Presione una tecla para continuar . . .

Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida\Computacion_Paralela_y_Distribuida\Computacion_Paralela_y_Distribuida.exe
- Output Size: 1,833,155,632,019,04 MiB
- Compilation Time: 2,58s

```

8) Se 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: $17 + 10 + 1992 = 2019 \Rightarrow 2 + 0 + 1 + 9 = 12 \Rightarrow 1 + 2 = 3$ lo que quiere decir que su número de Tarot es el 3.

Realizar un programa que pida una fecha, de la forma: **día mes año** donde día, mes y año son enteros, y dé como resultado el número de Tarot. El programa verificará si la fecha es correcta, esto es, los valores están dentro de los rangos permitidos.



```
D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida\Computacion_Paralela_y_Distribuida\1er_Corte...
----- CALCULADORA DE NUMERO DE TAROT -----

Ingresar la fecha de nacimiento separada por espacios, por ejemplo: 10 12 2001
29 06 1997
Su numero de Tarot es: 7
-----
Process exited after 7.911 seconds with return value 0
Presione una tecla para continuar . . .

Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida\Computaci
- Output Size: 1,83516120910645 MiB
- Compilation Time: 2,72s
```

Como se puede evidenciar el programa cumple el requerimiento solicitado, dado que satisface la obtención del número de tarot para una persona, dada su fecha de nacimiento digitada por el usuario, estableciendo una variable

para el día, el mes y el año, la cual se verifica por medio de un ciclo **while** que los números ingresados estén dentro de los rangos permitidos y generando el resultado por medio de dos funciones, una función llamada **sumFecha** la cual por medio de los parámetros **x**, **y** y **z** suma estas cifras y la función **numTarot** que se encarga de generar el número de tarot por medio de los parámetros de un entero suma y una variable **x**.

9) 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
 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
 2 3 . . . . .
  
```

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

```

1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Noveno Punto Taller
6  */
7
8  #include <iostream>
9  using namespace std;
10
11 //Función para imprimir triangulo
12 void printtriangle (int numfilas){
13     int i, j, k, z, espacios;
14
15     espacios = numfilas-1;
16
17     for(i = 1; i <= numfilas; i++){
18         for(j = 0; j < espacios; j++){
19             cout << " ";
20         }
21         for(k = i; k <= (i*2)-1; k++){
22             cout << k%10;
23         }
24         for(z = (i*2)-1; z > i; z--){
25             cout << (z-1)%10;
26         }
27         cout << endl;
28         espacios--;
29     }
30 }
31
32
33 // Se requiere implementar una aplicación que genere la secuencia de digitos mostrados
34 int main(int argc, char** argv) {
35     int numfilas;
36     do{
37         cout<<"Digite el numero de filas de la piramide, entre 11 y 20" << endl;
38         cin>>numfilas;
39     }while(numfilas < 11 || numfilas > 20);
40
41     printtriangle(numfilas);
42     return 0;
43 }
44
  
```

```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela
Digite el numero de filas de la piramide, entre 11 y 20
15

      1
     232
    34543
   4567654
  567898765
 67890109876
 7890123210987
 890123454321098
 90123456765432109
 0123456789876543210
 123456789010987654321
 23456789012321098765432
 3456789012345432109876543
 456789012345676543210987654
 56789012345678987654321098765

-----
Process exited after 2.653 seconds with return value 0
Presione una tecla para continuar . . .

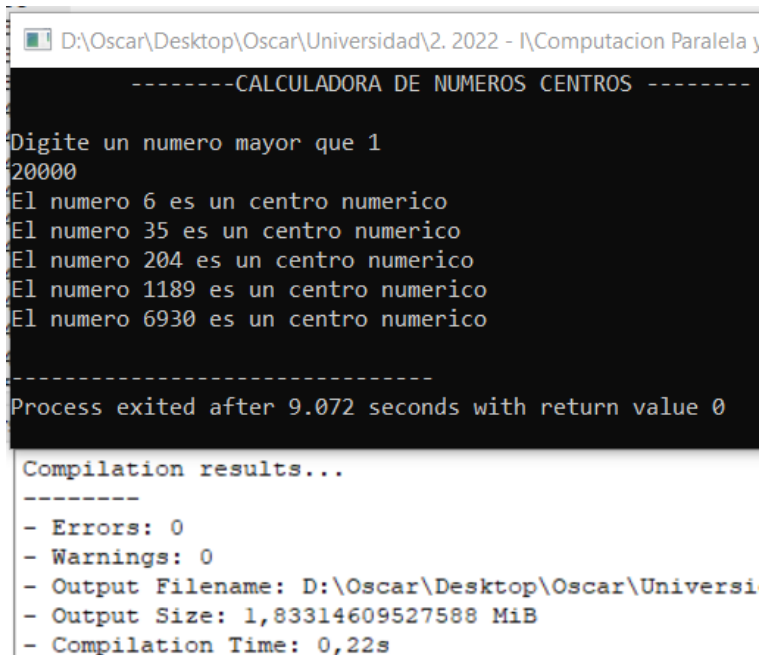
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Univers:
- Output Size: 1,83314609527588 MiB
- Compilation Time: 2,59s
  
```

Como se puede evidenciar el programa cumple el requerimiento solicitado, dado que satisface la obtención del triángulo por medio de un numero de filas digitado por el usuario y verificado por medio de un ciclo **while** que los

números ingresados estén dentro de los rangos permitidos (11 a 20) y haciendo uso de la función **printtriangle** la cual recibe por parámetro el numero de filas, lo construye y lo muestra, para este ejemplo un triángulo de 15 filas.

10) Un centro numérico es un número que separa una lista de números enteros (comenzando en 1) en dos grupos de números, cuyas sumas son iguales. El primer centro numérico es el 6, el cual separa la lista (1 a 8) en los grupos: (1, 2, 3, 4, 5) y (7, 8) cuyas sumas son ambas iguales a 15. El segundo centro numérico es el 35, el cual separa la lista (1 a 49) en los grupos: (1 a 34) y (36 a 49) cuyas sumas son ambas iguales a 595. Escribir un programa que calcule los centros numéricos entre 1 y n.

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Decimo Punto Taller
6  */
7
8  #include <iostream>
9  using namespace std;
10
11 //Funcion de busqueda de centros numéricos
12 void searchCenters(int n){
13
14 /*Se declaran variables para suma de números anteriores al centro y posteriores
15 al centro */
16 int sumBefore = 0, sumAfter = -1, center = 1;
17
18 do{
19     for (int i = 1; i < center; i++){
20         sumBefore += i;
21     }
22     for (int j = center+1; j <= center*2; j++){
23         if(sumBefore == sumAfter){
24             cout << "El numero " << center << " es un centro numerico" << endl;
25             break;
26         }else{
27             sumAfter += j;
28         }
29     }
30     center++;
31     sumBefore=0;
32     sumAfter=0;
33 }while (center <=n);
34
35
36
37 //Se requiere la implementación de una aplicación que calcule los centros numéricos entre 1 y n
38
39 int main(int argc, char** argv) {
40     int n;
41     cout << "\t -----CALCULADORA DE NUMEROS CENTROS ----- \n" << endl;
42     do{
43         cout << "Digite un numero mayor que 1" << endl;
44         cin >> n;
45     }while(n<1);
46     searchCenters(n);
47 }
```



```
D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida>g++ 10.cpp
-----CALCULADORA DE NUMEROS CENTROS -----
Digite un numero mayor que 1
20000
El numero 6 es un centro numerico
El numero 35 es un centro numerico
El numero 204 es un centro numerico
El numero 1189 es un centro numerico
El numero 6930 es un centro numerico

Process exited after 9.072 seconds with return value 0

Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Distribuida\10.cpp.out
- Output Size: 1,83314609527588 MiB
- Compilation Time: 0,22s
```

Como se puede evidenciar el programa cumple el requerimiento solicitado, dado que satisface la obtención de los números “centros” en un rango de 1 a un valor **n** digitado por el usuario y verificado por medio de un ciclo **while** y creando una función denominada **searchCenters** la cual recibe por parámetro el número digitado por el usuario, realiza la búsqueda a través de ciclos **for** y a medida que los va encontrando los va mostrando por consola.

11) 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

```
1  /*
2  Fecha: 09 de febrero de 2022
3  Autor: Oscar Julian Reyes Torres
4  Materia: Computación Paralela y Distribuida
5  Tema: Ejecucion Undecimo Punto Taller
6  */
7
8  #include <stdio.h>
9  #include <iostream>
10 #include <string>
11
12 using namespace std;
13
14 /*
15 Se requiereelaborar una aplicacion que calcule el importe a pagar para el caso de
16 una bicicleta, un moto, un vehiculo particular y un camion con un menú
17 */
18
19 main(int argc, char** argv) {
20     int opcion, km,tm,importe;
21
22     std::string menu= std::string("\t ----- SELECCIONE TIPO DE VEHICULO ----- \n")+
23         std::string(" 1. Bicicleta \n")+
24         std::string(" 2. Moto \n")+
25         std::string(" 3. Coche \n")+
26         std::string(" 4. Camion \n")+
27         std::string(" 5. Salir \n");
28
29     cout<<menu;
30     cin>>opcion;
31
32     while(opcion>5 || opcion<0){
33         cout<<"\n La opcion seleccionada no existe \n\n: "<<menu;
34         cin>>opcion;
35     }
36
37     switch(opcion) {
38         case 1 :
39             importe=100;
40             break;
41         case 2 :
42         case 3 :
43             cout<<"Digite la cantidad de Kilometros: ";
44             cin>>km;
45             importe=30*km;
46             break;
47         case 4:
48             cout<<"Digite la cantidad de Kilometros: ";
49             cin>>km;
50             cout<<"Digite la cantidad de Toneladas: ";
51             cin>>tm;
52             importe= 30 * km + 25 * tm;
53             break;
54         case 5:
55             return 0;
56
57     }
58     cout<<" \n El importe total a pagar es: $"<<importe<<"\n";
59
60 }
```

D:\Oscar\Desktop\Oscar\Universidad\2. 2022 - I\Computacion Paralela y Di

```
----- SELECCIONE TIPO DE VEHICULO -----
1. Bicicleta
2. Moto
3. Coche
4. Camion
5. Salir
4
```

Digite la cantidad de Kilometros: 100

Digite la cantidad de Toneladas: 30

El importe total a pagar es: \$3750

Process exited after 9.295 seconds with return value 0

Presione una tecla para continuar . . .

Compilation results...

```
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\Oscar\Desktop\Oscar\Universidad
- Output Size: 1,83689403533936 MiB
- Compilation Time: 2,94s
```

Como se puede evidenciar el programa cumple el requerimiento solicitado, dado que satisface la obtención del importe total a pagar para los diferentes medios de transporte como son una bicicleta, una moto, un coche y con camión, por medio de un menú de opciones la cual según sea la opción digitada por el usuario calcula un valor determinado, como el ejemplo anterior, en el cual era un camión con 100 kilómetros y una cantidad de 30 toneladas.