5.1. ¿Cuál es la salida del siguiente segmento de programa?

for (int cuenta = 1; cuenta < 5; cuenta++)

cout << (2 \* cuenta) << “ “;

respuesta

La salida del siguiente segmento de programa sería:

2 4 6 8

El bucle "for" se ejecuta mientras que la variable "cuenta" sea menor que 5. En cada iteración, se multiplica el valor de "cuenta" por 2 y se imprime en la salida estándar, seguido de un espacio. En este caso, el bucle se ejecuta cuatro veces, con los valores de "cuenta" siendo 1, 2, 3 y 4, respectivamente. Por lo tanto, la salida será "2 4 6 8".

5.2. ¿Cuál es la salida de los siguientes b ucles?

a) for (int n = 10; n > 0; n = n – 2)

{

cout << "Hola ";

cout << n << endl ;

}

b) for (double n = 2; n > 0; n = n - 0.5)

cout << m << " ";

respuesta

a) La salida del bucle sería:

Hola 10

Hola 8

Hola 6

Hola 4

Hola 2

El bucle comienza con `n` igual a 10. En cada iteración, se imprime "Hola" seguido del valor actual de `n`, y luego `n` se decrementa en 2. El bucle continúa mientras `n` sea mayor que 0. En cada iteración, `n` toma los valores 10, 8, 6, 4 y 2.

b) El código proporcionado tiene un error, ya que hace referencia a una variable `m` que no ha sido declarada ni inicializada. Presumiblemente, se supone que se utilizó una variable `n` en su lugar. Entonces, si corregimos el código para usar `n`, la salida del bucle sería:

2 1.5 1 0.5

El bucle comienza con `n` igual a 2. En cada iteración, se imprime el valor actual de `n` seguido de un espacio. Luego, `n` se decrementa en 0.5. El bucle continúa

5.3. Considerar el siguiente código de programa.

using namespace std;

int main(int argc, char \*argv[])

{

int i = 1 , n ;

cin >> n;

while (i <= n)

if ((i % n) == 0)

++i;

cout << i << endl;

system("PAUSE");

return EXIT\_SUCCESS;

}

a) ¿Cuál es la salida si se introduce como v alor de n, 0?

b) ¿Cuál es la salida si se introduce como v alor de n, 1?

c) ¿Cuál es la salida si se introduce como v alor de n, 3?

Resultado

El código corregido con algunas mejoras sería:

cpp

#include <iostream>

using namespace std;

int main(int argc, char \*argv[])

{

int i = 1, n;

cin >> n;

while (i <= n)

{

if ((i % n) == 0)

++i;

}

cout << i << endl;

system("PAUSE");

return 0;

}

a) Si se introduce como valor de `n` el número 0:

En este caso, el bucle `while` no se ejecutará ya que la condición `i <= n` no se cumple. Por lo tanto, la salida será:

1

b) Si se introduce como valor de `n` el número 1:

En este caso, el bucle `while` se ejecutará una vez porque la condición `i <= n` se cumple al inicio. Sin embargo, dentro del bucle, la condición `(i % n) == 0` no se cumple, por lo que no se incrementará el valor de `i`. La salida será:

1

c) Si se introduce como valor de `n` el número 3:

En este caso, el bucle `while` se ejecutará tres veces debido a la condición `i <= n`. Dentro del bucle, la condición `(i % n) == 0` se cumple cuando `i` es igual a 3. En ese caso, se incrementará el valor de `i` en 1 y continuará con la siguiente iteración. Al final, cuando `i` sea igual a 4, la condición del bucle `while` no se cumplirá y se imprimirá el valor de `i`, que es 4. La salida será:

4

5.4. Suponiendo que m = 3 y n = 5 ¿Cuál es la salida de los siguientes se gmentos de programa?

a) for (int i = 0; i < n; i++)

{

for (int j = 0; j < i; j++)

cout << “ \*”;

cout << endl;

}

b) for (int i = n; i > 0; i--)

{

for (int j = m; j > 0; j--)

cout << “ \* ”;

cout << endl;

}

a) La salida del segmento de programa sería la siguiente:

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

En este caso, se utiliza un bucle for anidado para imprimir una serie de asteriscos en forma de triángulo. El bucle exterior itera desde 0 hasta n-1, mientras que el bucle interior itera desde 0 hasta i-1 en cada iteración del bucle exterior. Esto asegura que se impriman el número correspondiente de asteriscos en cada línea.

b) La salida del segmento de programa sería la siguiente:

\* \* \*

\* \* \*

\* \* \*

En este caso, también se utilizan bucles for anidados para imprimir una serie de asteriscos. El bucle exterior itera desde n hasta 1, mientras que el bucle interior itera desde m hasta 1 en cada iteración del bucle exterior. Esto asegura que se impriman el número correspondiente de asteriscos en cada línea. Sin embargo, como m = 3, la salida consiste en tres asteriscos en cada línea y se repite n veces.

5.5. ¿Cuál es la salida de este bucle?

int i = 1 ;

while (i \* i < 10)

{

int j = i;

while (j \* j < 100)

{

cout << i + j << " ";

j \*= 2;

}

i++;

cout << endl;

}

cout << "\n\*\*\*\*\*\n";

La salida del bucle sería la siguiente:

2 3 5 9

4 5 7 11 8 9 11 15

6 7 9 13 10 11 13 17 12 13 15 19

El bucle exterior utiliza una condición `i \* i < 10` para determinar si debe continuar ejecutándose. Inicialmente, `i` se establece en 1.

Dentro del bucle exterior, se encuentra otro bucle while anidado. El bucle interior utiliza la condición `j \* j < 100` para determinar si debe seguir ejecutándose. Al inicio de cada iteración del bucle exterior, `j` se establece en el valor actual de `i`.

Dentro del bucle interior, se imprime `i + j` y luego `j` se multiplica por 2. Esto se repite hasta que `j \* j` sea mayor o igual a 100.

Después de cada iteración del bucle interior, se incrementa `i` en 1 y se imprime una nueva línea. El bucle exterior se repite hasta que `i \* i` sea mayor o igual a 10.

Finalmente, se imprime la línea "\*\*\*\*\*" fuera del bucle para marcar el final de la salida.