

Cp 4.5 - Control de flujo. Consolidación.
Curso 2024-2025



Para cada ejercicio, se espera que intentes predecir el resultado de la ejecución del código sin ejecutarlo en un entorno de desarrollo. Una vez que hayas hecho tus predicciones, puedes ejecutar el código para comprobar tus respuestas.

1. Dado el siguiente código:

```
int Calculate(int n)
{
    if (n < 5)
    {
        return n * 2;
    }
    else if (n < 10)
    {
        return n + 3;
    }
    else
    {
        return n - 2;
    }
}

int result1 = Calculate(3);
int result2 = Calculate(7);
int result3 = Calculate(12);
```

- a) ¿Qué valor devuelve Calculate(3)?
- b) ¿Qué valor devuelve Calculate(5)?
- c) ¿Qué valor devuelve Calculate(12)?

2. Dado el siguiente código:

```
int x = 10;
int y = 3;

if (x > 5)
{
    x += y;
}

if (x < 15)
{
    y += x;
}
else
```

```
{
    y -= x;
}
```

- a) ¿Cuál es el valor de x después del primer bloque if?
- b) ¿Cuál es el valor final de y?

3. Dado el siguiente código:

```
int x = 10;
int y = 5;
int result = 0;

if (x > y)
{
    result = x * y;
}
else
{
    result = x + y;
}

y = 0;

if (result > 20)
{
    result -= 10;
}
else
{
    result += 10;
}
```

- a) ¿Qué valor tiene result después del primer bloque if?
- b) ¿Qué valor tiene result después del segundo bloque if?

4. Dado el siguiente código:

```
int result = 0;

for (int i = 0; i < 5; i++)
{
    if (i % 2 == 0)
    {
        result += i;
    }
    else
    {
        result -= i;
    }
}
```

- a) ¿Cuál es el valor de sum después de la primera iteración?
- b) ¿Cuál es el valor de sum al final del bucle?

5. Dado el siguiente código:

```
string[] fruits = [ "apple", "banana", "cherry" ];
string message = "";
```

```

for (int i = 0; i < fruits.Length; i++)
{
    switch (fruits[i])
    {
        case "apple":
            message += "A";
            break;
        case "banana":
            message += "B";
            break;
        case "cherry":
            message += "C";
            break;
        default:
            message += "?";
            break;
    }
}

```

- a) ¿Cuál es el valor de message al final del bucle for?
- b) ¿Qué ocurre si se agrega .orange al arreglo fruits?

6. Dado el siguiente código:

```

int result = 1;

for (int i = 1; i <= 5; i++)
{
    if (i % 2 == 0)
    {
        result *= i;
    }
    else
    {
        result += i;
    }
}

```

- a) ¿Cuál es el valor de result después de la primera iteración?
- b) ¿Cuál es el valor de result al final del bucle?

7. Dado el siguiente código:

```

int a = 0;
int count = 0;

while (a < 10)
{
    a++;
    if (a % 2 == 0)
    {
        continue;
    }
    count++;
    if (count == 3)
    {
        break;
    }
}

```

- a) ¿Qué valor tiene a cuando el bucle se detiene?

b) ¿Cuántas veces se incrementa count?

8. Dado el siguiente código:

```
int sum = 0;

for (int i = 0; i < 3; i++)
{
    for (int j = 0; j < 2; j++)
    {
        sum += i + j;
    }
}
```

a) ¿Cuántas veces se ejecuta el bucle for interno?

b) ¿Cuál es el valor de sum al final del código?

9. Dado el siguiente código:

```
void Increment(int number)
{
    number += 5;
}

int originalNumber = 10;
Increment(originalNumber);
```

a) ¿Cuál es el valor de originalNumber antes de llamar al método Increment?

b) ¿Cuál es el valor de originalNumber después de llamar al método Increment?

c) ¿Por qué sucede esto?

10. Dado el siguiente código:

```
void ModifyArray(int[] numbers)
{
    numbers[0] = 99;
}

int[] originalArray = [1, 2, 3, 4];
ModifyArray(originalArray);
```

a) ¿Cuál es el valor de originalArray[0] antes de llamar al método ModifyArray?

b) ¿Cuál es el valor de originalArray[0] después de llamar al método ModifyArray?

c) ¿Por qué sucede esto?

11. Dado el siguiente código:

```
int Method1(int[] arr)
{
    int i = 0;
    int result = 0;
    while (i < 4)
    {
        result += arr[i++];
    }
    return result;
}

int Method2(int[] arr)
```

```

{
    int i = 0;
    int result = 0;
    while (i < 4)
    {
        result += arr[++i];
    }
    return result;
}

int[] myArray = [ 10, 20, 30, 40, 50 ];
int result1 = Method1(myArray);
int result2 = Method2(myArray);

```

- a) ¿Qué valor guarda result1?
- b) ¿Qué valor guarda result2?