

### **U3\_B4\_1:** ¿Qué pasaría si hubiéramos utilizado la forma de post-incremento `i++`?

Se imprimen números hasta el 11, ya que cuando `i` vale 10 y se entra en el bucle ocurre lo siguiente:

1. `System.out.println(i + " ")`: Se imprime el número 10
2. se evalúa la expresión `(i++<11)` para saber si debemos hacer una iteración más.  
En la condición anterior se hacen realmente dos cosas, incrementar una variable y comprobar una condición, el orden de estas operaciones es determinante en este caso. Al utilizar el operador de incremento `++` en modo postfijo ocurre:
  - ♦ primero: se comprueba si `i<11`, **es cierto**
  - ♦ segundo: se incrementa `i` y pasa a valer 11

como la condición es cierta se vuelve a entrar en el cuerpo del bucle, valiendo `i` 11 y se imprime el número 11.

### **U3\_B4\_2:**

```
import java.util.Scanner;
class Unidad3 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String opcion="";
        do{
            System.out.println("\n-----lista de actividades -----");
            System.out.println("comer");
            System.out.println("beber");
            System.out.println("acabar");
            System.out.print("Teclea tu acción preferida: ");
            opcion=sc.next();
            switch(opcion){
                case("comer"):
                    System.out.println("ñam ñam ñam ...");
                    break;
                case("beber"):
                    System.out.println("glu glu glu ...");
                    break;
                case("acabar"):
                    System.out.println("¡¡adios!!!");
                    break;
                default:
                    System.out.println("¡no existe esa opcion!");
            }
        }while(!opcion.equals("acabar"));
    }
}
```