

# PL2 Ingeniería del Software

## Ejercicio 2

Cristina Martínez Toledo: 09109126E

Roberto Seco Volkava: 09854422A

# Índice

<b>Código a analizar y numeración.....</b>	<b>2</b>
<b>Grafo obtenido.....</b>	<b>3</b>
<b>Cálculo de la complejidad y caminos básicos.....</b>	<b>4</b>

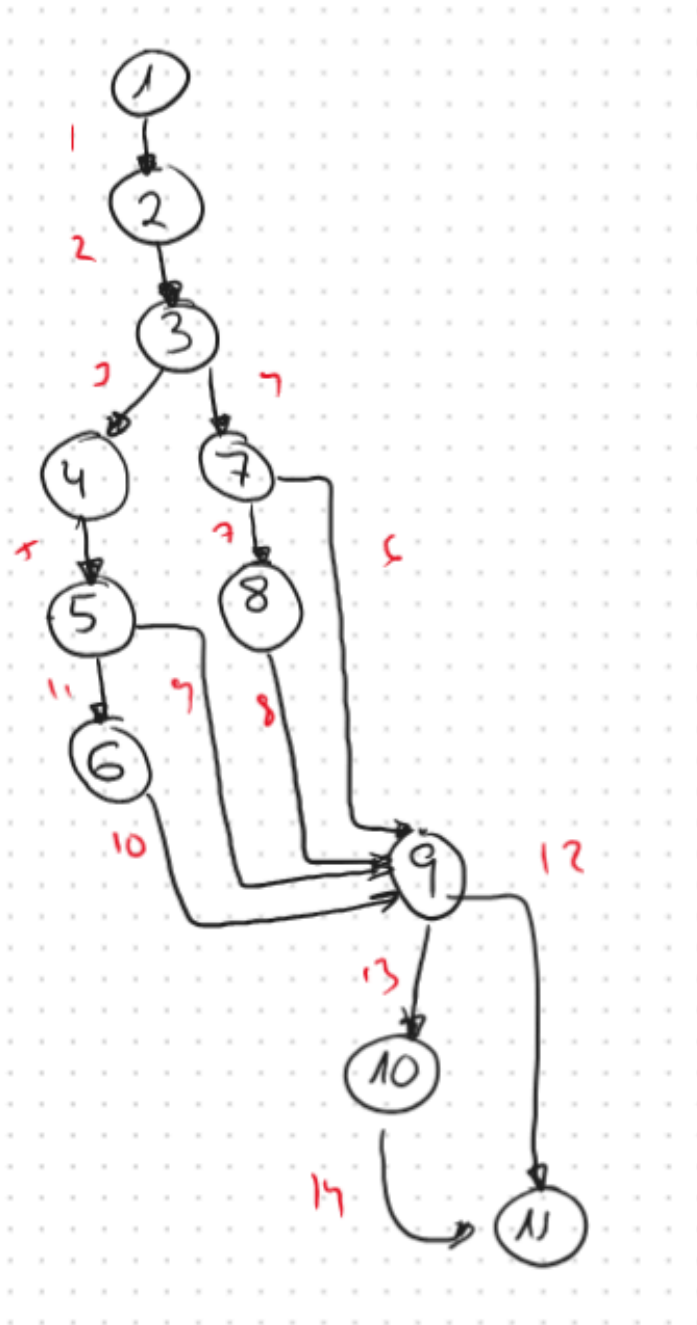
## Código a analizar y numeración

El código a analizar es el siguiente:

```
1 public int gestionInventario(int stock, int pedidos, boolean urgente, boolean proveedorActivo) {  
    int resultado = 0; 2  
    if (stock < 10 && pedidos > 0) { 3  
        resultado += pedidos; 4  
        if (urgente) { 5  
            resultado += 5; // margen de urgencia 6  
        }  
    } else if (stock >= 10) { 7  
        resultado -= 2; // pequeña penalización por sobrealmacenaje 8  
    }  
    if (!proveedorActivo) { 9  
        resultado = -1; // error en la gestión 10  
    }  
    return resultado; 11  
}
```

## Grafo obtenido

Según la numeración del código, representando cada “número” o parte como un nodo, este se puede expresar mediante el siguiente grafo:



## Cálculo de la complejidad y caminos básicos

Sabiendo que la complejidad ciclomática se puede obtener mediante la expresión

$V(G) = a - n + 2$ , donde  $a$  es el número de aristas y  $n$  el número de nodos. Utilizando el grafo obtenido, calculamos la siguiente suma:  $V(G) = 14 - 11 + 2 = 5$

**Por tanto, la complejidad ciclomática del código aportado es de 5**

Según el grafo obtenido, se puede ver que habría 8 caminos posibles:

1. Si stock es menor que 10 y pedidos es mayor a 0, entra en el primer if. Después, si es urgente, entra al segundo. Por último, si proveedorActivo=true, no entra en el último if. El resultado será el valor de pedidos+5
2. Si stock es menor que 10 y pedidos es mayor a 0, entra en el primer if. Después, si es urgente, entra al segundo. Por último, si proveedorActivo=false, entra en el último if. El resultado será -1
3. Si stock es menor que 10 y pedidos es mayor a 0, entra en el primer if. Después, si no es urgente, no entra al segundo. Por último, si proveedorActivo=true, no entra en el último if. El resultado será el valor de pedidos
4. Si stock es menor que 10 y pedidos es mayor a 0, entra en el primer if. Después, si no es urgente, no entra al segundo. Por último, si proveedorActivo=false, entra en el último if. El resultado será -1
5. Si stock es mayor o igual a 10, entra en el elif. Por último, si proveedorActivo=true, no entra en el último if. El resultado será -2
6. Si stock es mayor o igual a 10, entra en el elif. Por último, si proveedorActivo=false, entra en el último if. El resultado será -1
7. Si stock es menor a 10 pero pedidos no es mayor a 0, no entra ni en el primer if ni en el elif. Por último, si proveedorActivo=true, no entra en el último if. El resultado será 0
8. Si stock es menor a 10 pero pedidos no es mayor a 0, no entra ni en el primer if ni en el elif. Por último, si proveedorActivo=false, entra en el último if. El resultado será -1

Con estos caminos se puede observar en los casos 2, 4, 6 y 8, que debido al último "if", común a todos los caminos, si proveedorActivo tiene valor false, el resultado siempre será -1.

Debido a esto, se podría reducir a solamente **5 caminos básicos** (1,3,5,7 y cualquiera de los mencionados en los que el valor final sea -1), lo que coincide con la complejidad ciclomática obtenida.