

# Temario para examen

## Complejidad Ciclomática

La **complejidad ciclomática** mide el número de **caminos independientes** en un **grafo de flujo** de un programa.

✨ **Un camino independiente** es cualquier secuencia de ejecución que introduce, al menos, un nuevo conjunto de sentencias.

### Fórmulas para calcularla

Método	Fórmula	Descripción
♦ <b>Aristas y nodos</b>	$V(G) = \frac{A - N + 2}{2}$	A = aristas, N = nodos
♦ <b>Regiones cerradas</b>	$V(G) = R$	R = número de regiones cerradas en el grafo
♦ <b>Nodos de condición</b>	$V(G) = C + 1$	C = cantidad de nodos de decisión ( <i>if, while, for, etc.</i> )

## Creación de Tests con Maven

### Pasos para crear un proyecto en Maven

1 Ejecutar en la terminal:

```
mvn archetype:generate
```

2 Presionar **Enter** hasta que solicite:

- **groupId** → **{nombre del paquete principal}**
- **artifactId** → **{nombre del proyecto}**

## SonarQube con Docker

### Instalación y ejecución en Ubuntu

1 **Instalar Docker** → [Guía oficial](#)

## 2 Descargar la imagen de SonarQube

```
sudo docker pull sonarqube:lts-community
```

## 3 Crear el contenedor en el puerto 9000

```
sudo docker create -i -t -p 9000:9000 --name sonar sonarqube:lts-community
```

## 4 Iniciar SonarQube

```
sudo docker start --attach -i sonar
```

## 5 Acceder desde el navegador → <http://localhost:9000>

- Usuario y contraseña por defecto: **admin:admin** (ya cambiada)

## 6 Crear un proyecto en local

- Si seleccionas **Maven**, SonarQube generará un comando que debes ejecutar en la terminal.

### Otra forma de integración con Maven

Modificar el archivo **POM.xml** añadiendo las siguientes propiedades:

XML

```
<properties>
  <sonar.projectKey>demo</sonar.projectKey>
  <sonar.host.url>http://localhost:9000</sonar.host.url>
</properties>
```

---

## Token de autenticación ejercicio de clase

```
sqp_8fa1c5a66d5f48c0ae0b2aa5ca2fd7a5f755f955
```