

#### Objetivos.

- 1) Familiarizarse con el uso de diversas herramientas para la realización de pruebas.
- 2) Adquirir ciertos conocimientos básicos sobre técnicas de prueba.

#### Descripción.

Disponemos de la siguiente clase *Triangulo.java*<sup>1</sup>, que sirve para determinar el tipo de un triángulo en función de las longitudes de sus tres lados.

```
package com.iso2.dominio;

public class Triangulo
{
    protected int i, j, k;
    protected int tipo;
    public static final int ESCALENO=1, ISOSCELES=2, EQUILATERO=3, NO_TRIANGULO=0;

    public Triangulo()
    {
    }

    public Triangulo(int i, int j, int k)
    {
        this.i=i;
        this.j=j;
        this.k=k;
    }

    public void setI(int v) throws IllegalArgumentException {
        if (v<=0) throw new IllegalArgumentException();
        i=v;
    }

    public void setJ(int v) throws IllegalArgumentException {
        if (v<=0) throw new IllegalArgumentException();
        j=v;
    }

    public void setK(int v) throws IllegalArgumentException {
        if (v<=0) throw new IllegalArgumentException();
        k=v;
    }

    public int getTipo() {
        if (i==j) { tipo=tipo+1; }
        if (i==k) { tipo=tipo+2; }
        if (j==k) { tipo=tipo+3; }

        if (i<=0 || j<=0 || k<=0) {
            tipo=Triangulo.NO_TRIANGULO;
            return tipo;
        }
        if (tipo==0)
```

---

<sup>1</sup> El código está inspirado en el propuesto por Myers (1990) en *The art of software testing*.

```

{
    if (i+j<=k || i>=j+k || i+k>=j) {
        tipo=Triangulo.NO_TRIANGULO;
        return tipo;
    } else {
        tipo=Triangulo.ESCALENO;
        return tipo;
    }
}
if (tipo>3) {
    tipo=Triangulo.EQUILATERO;
    return tipo;
} else if (tipo==1 && i+j>k)
{
    tipo=Triangulo.ISOSCELES;
    return tipo;
} else if (tipo==2 && i+k>j)
{
    tipo=Triangulo.ISOSCELES;
    return tipo;
} else if (tipo==3 && j+k>i)
{
    tipo=Triangulo.ISOSCELES;
    return tipo;
} else {
    tipo=Triangulo.NO_TRIANGULO;
    return tipo;
}
}
}

```

En esta práctica construiremos casos de prueba para comprobar que la clase *Triangulo.java* (y, en particular, el método *getTipo*) realiza adecuadamente su cometido.

Herramientas que utilizaremos:

- 1) JUnit. Puede descargarse gratuitamente de [www.junit.org](http://www.junit.org). No obstante, viene incluido como plugin en Eclipse.
- 2) Coverlipse. Puede descargarse gratuitamente de <http://coverlipse.sourceforge.net/index.php>. Se trata de un plugin para Eclipse que utiliza los mismos casos de prueba que JUnit pero que, además, ofrece como resultados ciertas medidas de cobertura.
- 3) Koalog Code Coverage. Se trata de una herramienta para pruebas de caja blanca de Koalog. Se puede descargar una versión de evaluación de <http://www.koalog.com/php/kover.php>; no obstante, tenemos licencia académica.

El proceso de pruebas que seguiremos puede representarse gráficamente de la siguiente forma:

