

## Examen de Programación (Grado en Física, Grado en Matemáticas)

Septiembre 2011

**Primera parte (5 puntos, 50% nota del examen).**

- 1) Se desea obtener el valor de la siguiente serie para distintos valores del número de términos calculados:

$$S = \sqrt{\frac{2}{1} + \frac{3}{2} + \frac{2}{3} + \frac{3}{4} + \frac{2}{5} + \frac{3}{6} + \frac{2}{7} + \frac{3}{8} + \frac{2}{9} + \frac{3}{10} + \frac{2}{11} + \dots}$$

Para ello se deberá implementar un método estático llamado `valorSerie` dentro de una clase llamada `Serie`. El método recibirá como parámetro el número de términos a calcular y retornará el valor de la serie para el número de términos indicado.

- 2) Se dispone de la clase `Partícula` que permite calcular la velocidad con la que se mueve una partícula. El código de dicha clase (sólo las partes relevantes para el problema) es:

```
public class Partícula
{
    ... // atributos y métodos no relevantes para el problema

    /**
     * Calcula la velocidad exacta de la partícula
     * @return velocidad exacta de la partícula
     * @throws ErrorVelocidad cuando no es posible realizar el
     * cálculo
     */
    public double calculaVelocidadExacta() throws ErrorVelocidad {
        ... // código no relevante para el problema
    }

    /**
     * Calcula la velocidad aproximada de la partícula
     * @return velocidad aproximada de la partícula
     */
    public double calculaVelocidadAproximada() {
        ... // código no relevante para el problema
    }
}
```

Escribe un método de la misma clase `Partícula` que verifique el comportamiento descrito por el siguiente comentario de documentación:

```
/**
 * Calcula la velocidad de la partícula.
 * @return velocidad exacta en el caso de que sea posible realizar
 * su cálculo. Si no es posible retorna la velocidad aproximada.
 */
```

- 3) Se pretende procesar ficheros que contienen medidas de tiempo, espacio y velocidad. Cada medida está formada por su valor (un número real) y la unidad en la que se mide (s, m, m/s). Hay un espacio entre el valor y la unidad de cada medida y también hay un espacio entre las distintas medidas. Las medidas de las distintas magnitudes están distribuidas aleatoriamente en el fichero. Un ejemplo de fichero podría ser:

```
10.5 m 4.3 t -3.45 m/s 6.0 s 0.1 m/s 7.2 s 8.4 s
14.8 m 2.3 m/s 1.8 m/s 22.3 m 0.5 m/s 8.9 s
```

Escribir una clase que contenga un método estático `maxVelocidadEnFich` que recibe como parámetro el nombre de un fichero como el descrito anteriormente y retorna el mayor valor de velocidad que contiene.

Por ejemplo, para el fichero mostrado, el método debería retornar el valor 2.3.

- 4) Completa el método que aparece a continuación:

```
/**
 * Cuenta el número de elementos de una matriz que son iguales a un
 * valor dado.
 * @param matriz array bidimensional con los elementos de la matriz
 * @valor valor con el que comparar los elementos de la matriz
 * @return número de elementos de la matriz que son iguales al
 * valor indicado
 */
public static int cuentaOcuurrencias(int[][] matriz, int valor) {
    ... // HACER POR EL ALUMNO
}
```