

Prueba 3 de Evaluación Continua

Análisis y Diseño de Software (2010/2011)

Contesta en esta misma hoja.

Apellidos:
Nombre:

Apartado 1. (4 puntos)

Completa el siguiente programa, para que su ejecución resulte en la siguiente salida:

```
{Castejon, Cuenca (2006)=0.2, Alcobendas, Madrid (2008)=107.51, Alcobendas, Madrid (2009)=106.0, Castejon, Navarra (2006)=3.9, Abezames, Zamora (2001)=0.107}
```

```
import _____; // Complétalo

class Municipio _____ { // Complétalo
    private String provincia, municipio;
    private int añoCenso;

    public Municipio(String p, String m, int a) {
        this.provincia = p;
        this.municipio = m;
        this.añoCenso = a;
    }
    // Completa el código (a continuación) ...

} // end class Municipio

public class Colecciones {
    public static void main(String[] args) {
        _____ poblacionMunicipio = new _____(); // Complétalo
        // Provincia, Municipio, AñoCenso y Población en miles
        poblacionMunicipio.put( new Municipio("Madrid", "Alcobendas", 2009), 106.0);
        poblacionMunicipio.put( new Municipio("Madrid", "Alcobendas", 2008), 107.51);
        poblacionMunicipio.put( new Municipio("Zamora", "Abezames", 2001), 0.107);
        poblacionMunicipio.put( new Municipio("Cuenca", "Castejon", 2006), 0.2);
        poblacionMunicipio.put( new Municipio("Navarra", "Castejon", 2006), 3.9);

        System.out.println(poblacionMunicipio);
    }
}
```


Prueba 3 de Evaluación Continua

Análisis y Diseño de Software (2010/2011)

Contesta en esta misma hoja.

Apellidos:

Nombre:

Apartado 2. (3.5 puntos)

Se quiere construir una clase que seleccione los elementos de un Conjunto (java.util.Set) obteniendo el subconjunto formado por los elementos que cumplen cierta condición. La condición se da mediante un objeto que implementa la interfaz genérica Seleccion.

Completa el siguiente programa, para que su ejecución resulte en la siguiente salida:

Subconjunto seleccionado: [0, 2, 5, 13]

```
import java.util.Set;
import java.util.TreeSet;
import java.util.Arrays;

interface Seleccion<T> {
    public boolean condicion(T x);
}
// Definimos una clase Seleccion para seleccionar los numeros no negativos.
class EliminaNegativos _____ { // Complétalo
    // Completa el código (a continuación) ...

} // end class EliminaNegativos

class SubconjuntoSeleccionado _____ { // Complétalo
    // Completa el código (a continuación) ...

} // end class SubconjuntoSeleccionado

public class Genericos {
    public static void main(String[] args) {
        Set<Integer> set = new TreeSet<Integer>(Arrays.asList(2, -1, 0, -9, 13, 5));
        SubconjuntoSeleccionado<Integer> s =
            new SubconjuntoSeleccionado<Integer>(set, new EliminaNegativos());
        System.out.println("Subconjunto seleccionado: " + s);
    }
}
```


Prueba 3 de Evaluación Continua

Análisis y Diseño de Software (2010/2011)

Contesta en esta misma hoja.

Apellidos:

Nombre:

Apartado 3. (2.5 puntos)

¿Qué salida produce la ejecución del siguiente programa?

```
class ExcepcionNumerica extends Exception {
    public ExcepcionNumerica () { System.out.println("Creada Excepcion Numerica"); }
}
class ExcepcionFibonacci extends ExcepcionNumerica {
    public ExcepcionFibonacci (String s) { System.out.println("Excepcion Fibonacci: " + s); }
}

public class ManejaExcepciones {
    private static void imprimeFibonacci(int[] nums) {
        int i = 0;
        try {
            for (int n : nums) {
                System.out.println("Fibonacci(" + n + ") = " + fibonacci(n));
                i++;
            }
        } catch (ExcepcionFibonacci e){
            System.out.println("en Fibonacci(" + nums[i] + ").");
        } catch (ExcepcionNumerica e) {
            System.out.println("Error en Fibonacci(" + nums[i] + ").");
        } finally {
            System.out.println("Se han imprimido " + i + " fibonacci.");
        }
    }

    private static int fibonacci(int n) throws ExcepcionNumerica {
        if (n < 0) throw new ExcepcionNumerica();
        if (n > 46) throw new ExcepcionFibonacci("resultado demasiado grande para int");
        if (n == 0) return 0;
        if (n == 1) return 1;
        return fibonacci(n-1) + fibonacci(n-2);
    }

    public static void main(String[] args) throws Exception {
        int[] v = {3, -5, 1, 16};
        int[] w = {0, 50, -2, -8};
        imprimeFibonacci(v);
        imprimeFibonacci(w);
    }
}
```

Salida esperada: