POLITÉCNICA

Escuela Técnica Superior de Ingeniería de Sistemas Informáticos
Universidad Politécnica de Madrid

Examen de Programación Orientada a Objetos (Plan 2014) — **29 de octubre de 2018**

Nº matrícula:_____ **Grupo:**_____ **Nombre:** _____

Apellidos: _____

Problema 1. (2 *puntos*)

Completa la implementación del método `union` de la clase `Conjunto` vista en la práctica del tema 1:

```
public class Conjunto implements IConjunto {
   private boolean[] conjunto;
   public Conjunto(){
      this.conjunto = new boolean[SIZE];
   }
   public boolean get(int posicion) {
       if(posicion<SIZE) return conjunto[posicion];
        return false;
   }
   public void set(int posicion, boolean valor) {
     if(posicion<SIZE) conjunto[posicion] = valor;
   }
   public IConjunto union (Iconjunto c){
     Conjunto auxiliar = new Conjunto();
     for (int i = 0; i < SIZE; i++)
        auxiliar.set(i, c.get(i) || conjunto[i]);
     return auxiliar;
   }
}
```

Problema 2. (*2 puntos*).

En la práctica del tema 3, se definían las interfaces `ISemaforo` e `ISemaforoDcha`, y las clases `Semaforo` y `SemaforoDcha`. Escribe las cabeceras/declaración de `ISemaforoDcha` y de `SemaforoDcha`.

```
public interface ISemaforoDcha extends ISemaforo
public class SemaforoDcha extends Semaforo implements ISemaforoDcha
```

y el método `equals` de la clase SemaforoDcha (2 Semáforos son iguales si sus luces coinciden)

```
public boolean equals(ISemaforoDcha s) {
   return super.equals(s) && colorDcha == s.getColorDcha();
}
```

Problema 3. (*4 puntos*).

En la práctica del tema 6, "Juego de la vida", completa el método completo donde se modifica el tablero. La clase `Tablero` proporciona el método `int getCasilla(int x, int y)`, y contiene la propiedad `int[][] tablero`.

```
public void set(Tablero tablero) {
  supportTablero.firePropertyChange("tablero", this. tablero, tablero);
  for (int x=0;x<tablero.size();x++)
       for (int y=0;y<tablero.size();y++)
         this.tablero[x][y] = tablero.getCasilla(x,y);
}
```

Completa la cabecera de la clase `Vista` y el método que la actualiza:

```
public class Vista extends Canvas implements PropertyChangeListener {

  public void propertyChange(PropertyChangeEvent evt) {
      tablero = (Tablero) evt.getNewValue();
      repaint();
  }
}
```

Problema 4. (*2 puntos*).

En la práctica de excepciones del tema 5, escribe la porción de código que lanza una excepción cuando el tamaño de la matrícula no es de 7 caracteres o cuando el primer carácter no es un dígito.

```
public static void valida(String matricula) throws ExMatricula2 {
     if (matricula.length()!=7)
         throw new ExMatricula2(ExMatricula2.Fallo.LONGITUD);
     else {
       if (!Character.isDigit(matricula.charAt(0)))
          throw new ExMatricula2(ExMatricula2.Fallo.NUMERO_UNO);
     }
}
```