

Nombre y Apellidos: _____

Hojas entregadas aparte: _____

Test. (1,5 puntos) Solo hay una respuesta correcta en cada pregunta. Marcar de forma clara la respuesta correcta (una X al lado, por ejemplo). Las respuestas en blanco no quitan puntuación. Las respuestas incorrectas quitan un tercio de la puntuación de las correctas.

1. Si queremos conservar el orden de inserción en un Map, usaremos:
 1. LinkedHashMap
 2. TreeMap
 3. HashMap
 4. HashSet
2. Si queremos utilizar una cola LIFO (Pila), no usaremos Stack porque está obsoleta. Usaremos:
 1. LinkedHashMap
 2. Deque
 3. ArrayList
 4. Vector
3. Las dos operaciones básicas que se realizan con una Pila (Stack) son:
 1. push y pop
 2. add y remove
 3. peek y pop
 4. put y remove
4. Si queremos crear un fichero en Java sin contenido (como el touch de UNIX), usaremos:
 1. File fichero=new File("Ruta");
 2. fichero.createNewFile();
 3. Files.touch("Ruta");
 4. Un fichero debe tener contenido obligatoriamente
5. Si afirmamos que una clase solo debe tener un motivo para cambiar, nos referimos a:
 1. El principio de responsabilidad única
 2. El principio KISS
 3. Polimorfismo
 4. El principio DRY
6. Si definimos un método de una clase como abstract
 1. Las clases que hereden de esa, implementarán obligatoriamente ese método
 2. La propia clase debe definirse como abstract
 3. Esa será la única clase del fichero java
 4. El método no será visible para las clases que hereden de esa clase

Nombre y Apellidos: _____

Ejercicios Prácticos. Como criterios de calificación de los ejercicios, se seguirán los mismos que se han tratado en clase (la ejecución correcta de un código sólo implica un 5 en la nota) Se valorará: Calidad de los comentarios (si éstos fueran necesarios). Que los nombres de Clases, atributos y métodos sean adecuados. Que los modificadores de acceso sean adecuados. La claridad del código. La eficiencia. El tratamiento de errores. Que la salida del programa sea clara, etc. **Cualquier aspecto no especificado en el enunciado será resuelto según el criterio del alumno. Únicas abreviaturas admitidas: psvm y sout**

1. (4 puntos) (1 leer el fichero de texto, 2 puntos construir el Map, 1 puntos escribir el Map en un fichero) Tenemos un fichero de texto "países.txt" con el siguiente formato:

España

Castellano, Gallego, Euskera, Catalán

Canadá

Francés, Inglés

Filipinas

Inglés, Castellano

.....

Queremos leer el fichero y guardarlo en un fichero de objetos (un solo objeto, un Map<String,List<String>>). Implementar un programa que realice la tarea pedida.

2. (1,5 puntos) (0,5 puntos leer el fichero, 1 punto imprimirlo en pantalla) Leer el fichero creado en el punto anterior e imprimirlo en pantalla

3. (2 puntos) (1 punto comportamiento correcto, 1 punto no repetir código) Completar el siguiente programa de modo que se debe reflejar en el texto de las etiquetas el número de veces que el usuario ha hecho clic en cada uno de los botones.

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class Botones {
    static JFrame frame;
    static JButton botonArriba=new JButton("Arriba");
    static JButton botonAbajo=new JButton("Abajo");
    static JLabel etiqueta=new JLabel();
    static int pulsadasArriba=0;
    static int pulsadasAbajo=0;
    public static void main(String[] args) {
        frame = new JFrame("DOS BOTONES"); //TITULO
        frame.setResizable(false);
        frame.setLayout(new GridLayout(3, 0));
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); //CERRAR VENTANA
        frame.add(etiqueta);
        frame.add(botonArriba);
        frame.add(botonAbajo);
    }
}
```

Nombre y Apellidos: _____

```

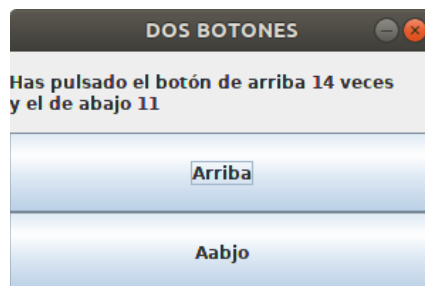
    botonArriba.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent actionEvent) {
            //A rellenar por el alumno
        }
    });
    botonAbajo.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent actionEvent) {
            //A rellenar por el alumno
        }
    });
    frame.setSize(300, 200);
    frame.setLocationRelativeTo(null);
    frame.setVisible(true); //MUESTRA LA VENTANA
}
}

```

Ventana Inicial:



Ventana después de pulsar varias veces los botones:



4. (1 punto) Disponemos de un archivo de acceso aleatorio puntos.dat con la información sobre puntos en el plano (pares x,y de enteros). Desarrollar un método que modifique el valor del punto en una determinada posición del fichero. Por ejemplo, la llamada:

punto(4,65,32)

pone en el cuarto registro de puntos los enteros 65 y 32 uno a continuación del otro.

El fichero se supone que no está abierto y es siempre el mismo "puntos.dat"

La signature del método es:

```

private static void punto(int posicion, int x, int y) throws
IOException {
    //Rellenar por el alumno
}

```