Examen de Fundamentos de Informática. 9/09/ 2011





Centro Politécnico Superior Universidad de Zaragoza

Duración total del examen: 3 horas

Ejercicio 1 [3,0 puntos]

Los profesores de la asignatura de Fundamentos de Informática han almacenado las notas de prácticas de todos los alumnos matriculados en un fichero de texto llamado "notas.txt". Cada línea del fichero corresponde con las notas e información de un único alumno. Más concretamente, cada línea consta de: una secuencia de ocho números reales, con un decimal de precisión y un valor comprendido en el rango [0,0..10,0], que representan la nota de cada una de las prácticas realizadas; y, el nombre del alumno. Por ejemplo, la siguiente línea representa las calificaciones obtenidas por Pedro Ferrer:

6,0 4,0 7,0 6,0 8,5 4,0 5,0 3,5 Pedro Ferrer

El fichero está libre de errores y los profesores conocen el número de alumnos matriculados en la asignatura, es decir, el número de líneas del fichero.

Por otro lado, los profesores de la asignatura han programado en Java la clase Alumno. Un objeto de esta clase almacena el resultado de evaluar las prácticas de un alumno, concretamente, el nombre completo del alumno, su nota media de prácticas y, finalmente, si ha superado las prácticas (si la nota media es mayor o igual a 5,0). Para almacenar esta información, la clase consta de tres atributos privados: _nombre, _nota y _superada. Además, la clase ofrece un constructor y los correspondientes métodos SET y GET.

```
public class Alumno {
    private String _nombre;
    private double _nota;
    private boolean _superada;

    public Alumno(String nb, boolean sp, double nt){}
    public void setNombre(String nb){}
    public void setNota(double nt){}
    public void setSuperada(boolean sp){}
    public String getNombre(){}
    public double getNota(){}
    public boolean getSuperada(){}
}
```

Los profesores quiere procesar la información del fichero de texto y almacenarla en una estructura arreglo de tipo Alumno []. Se creará un objeto Alumno con la información de cada alumno almacenada en el fichero. La nota media de un alumno será la media aritmética de las notas parciales de cada una de las prácticas. Una vez conocida esta nota media, se podrá determinar el valor del atributo _superada. Por tanto, SE PIDE programar en Java el siguiente método:

```
public static Alumno[] calculoNotasAlumnos (String fileName, int num alumnos)
```

Los parámetros de entrada del método son el nombre de un fichero de texto con el formato anterior (fileName) y el número de alumnos (o filas) del fichero (num_alumnos). El método debe devolver como resultado la estructura de tipo arreglo Alumno [] con la información procesada de los alumnos.

Ejercicio 2 [3,5 puntos]

Los profesores de la asignatura de Fundamentos de Informática quieren poder disponer de una representación visual que muestre los resultados obtenidos por los alumnos en las prácticas. Utilizando el método implementado en el apartado anterior, se puede calcular una estructura de tipo Alumno [] que contiene la nota final obtenida

por cada alumno y si superó finalmente las prácticas. El objetivo de este ejercicio es representar por pantalla un histograma de notas a partir de la información contenida en la estructura de notas finales.

El histograma que debe representarse constará de una fila por cada rango de calificación: Sobresaliente (notas comprendidas en el rango [9,0..10,0]), Notable (entre [7,0..9,0)), Aprobado (entre [5,0..7,0)) y No Superadas ([0,0..5,0)). Para cada rango concreto del histograma se escribirá tantos caracteres "*" como alumnos hayan obtenido una nota media contenida en el rango. Por ejemplo, el siguiente histograma representa que 3 alumnos obtuvieron la calificación final de Notable, 6 la calificación de Aprobado y, 9 no superaron las prácticas. Observad que es posible encontrar rangos que no tengan alumnos, como es el caso del correspondiente a la calificación de Sobresaliente.

```
Sobresaliente:
Notable: ***
Aprobado: ******
No superadas: *******
```

SE PIDE programar en Java el siguiente método:

```
public static void verHistogramaClase(Alumno[] clase)
```

El parámetro único de entrada del método es la estructura de tipo arreglo que contiene las notas medias de todos los alumnos matriculados en la asignatura. Suponemos que el contenido de esta estructura ha sido correctamente calculado y almacenado con el método del apartado anterior.

Ejercicio 3 [3,5 puntos]

Escribir un programa Java que permita cifrar y descifrar ficheros de texto. El método utilizado para cifrar los mensajes tendrá como parámetros de entrada el nombre del fichero original (ficheroOrigen), el nombre del fichero resultado (ficheroDestino), y un número entero (clave). A partir del número entero se generará el crápulo de dicho número y se generará un fichero en el que se sustituyen todos los caracteres por el correspondiente carácter resultante de sumar al código ASCII del número el crápulo del número utilizado como clave. Se respetará la estructura de líneas del fichero original no cifrando los finales de línea.

El mismo método recibe como parámetro adicional un argumento de tipo booleano (cifrado) que indica si el fichero origen se debe cifrar o descifrar (concretamente, valor TRUE para cifras y FALSE para el proceso contrario). En el caso de que se cifre se suma el crápulo al código ASCII, y en caso contrario se resta.

NOTA: Dado un número entero positivo, su crápulo es el número que se obtiene de la siguiente forma: se suman los dígitos que lo componen. Si el valor de la suma es menor de 10, el crápulo es el valor obtenido; si no, el crápulo es el crápulo de la suma de los dígitos (ej. crapulo(13)=4, crapulo(492)= 6, crapulo(5678)=8)

Para simplificar supondremos que el texto a cifrar sólo contiene caracteres mayúsculas (sin acentos), dígitos y signos de puntuación "," y ".". De esta forma el código ASCII a cifrar y descifrar está dentro del código ASCII (0-127).

SE PIDE programar en los siguientes métodos: