



**Semana 6: Bucles**

**Programación**

**2018 - 2019**

**Grado en Ingeniería Informática**

**Ejercicio 1.** Escribir un programa que lea números enteros por teclado hasta que el usuario introduzca el 0. El programa debe imprimir por pantalla la suma de todos los números y la media (con decimales), sin incluir el 0. Se debe considerar el caso en el que el usuario introduzca solamente un 0 (en ese caso la media será 0 y la suma también 0). La salida debe ser:

La suma de los números es: 10.0

La media de los números es: 2.5

**Ejercicio 2.** Escribir un programa que imprima por pantalla todos los números flotantes del 1 al 100, separados por comas, sumando de 1.5 en 1.5. Ejemplo de salida: 1.0, 2.5, 4.0, 5.5, ...100.0

**Ejercicio 3.** Escribir un programa que reciba por teclado un número entero y cuántos múltiplos hay que escribir de ese número por pantalla. Ejemplo, si recibe un 3 y un 5, deberá imprimir 3, 6, 9, 12, 15.

**Ejercicio 4.** Escribir un programa para jugar al juego de acertar un número. El programa generará automáticamente un número entre 1 y 100 y le pedirá al usuario que intente adivinarlo. Cada vez que el usuario lo intente, le dirá si su número es mayor o menor que el buscado. Al final debe imprimir qué número era y cuántos intentos ha necesitado.

**Ejercicio 5.** Escribir un programa que genere e imprima los N números perfectos menores de un cierto número introducido por teclado. Un número es perfecto si es igual a la suma de todos sus divisores positivos sin incluir el propio número. Por ejemplo el número 6 es perfecto ya que tiene como divisores: 1, 2, 3 y 6 (pero el 6 no se cuenta como divisor para comprobar si es perfecto). Si sumamos  $1 + 2 + 3 = 6$ . Ejemplo:

Indica el límite superior para generar números perfectos y pulsa Enter:

10000

El número 6 es perfecto

El número 28 es perfecto

El número 496 es perfecto

El número 8128 es perfecto

**Ejercicio 6.** Escribir una clase que implemente el siguiente juego: tres jugadores lanzan cada uno una moneda al aire. Ello se simulará con un generador aleatorio que dé dos posibles valores: uno simulará que ha salido cara y el otro que ha salido cruz. Si las tres monedas caen en CARA o CRUZ no gana nadie. Gana el jugador cuya moneda sea diferente a las del resto. La clase debe simular el lanzamiento de las tres monedas y debe imprimir por pantalla lo que ha sacado cada jugador y el ganador del juego en caso de que exista. La partida se realizará por un total de 10 tiradas. El ganador final será el jugador que haya obtenido más puntos en las 10 tiradas. La salida por pantalla debe ser como la siguiente:

El resultado de la tirada 1 es:  
El jugador 1 ha sacado CRUZ  
El jugador 2 ha sacado CRUZ  
El jugador 3 ha sacado CARA  
Ha ganado el jugador 3 en la tirada 1

EL RESULTADO HASTA EL MOMENTO ES:  
JUGADOR 1 --> 0 PUNTOS  
JUGADOR 2 --> 0 PUNTOS  
JUGADOR 3 --> 1 PUNTOS

El resultado de la tirada 2 es:  
El jugador 1 ha sacado CRUZ  
El jugador 2 ha sacado CARA  
El jugador 3 ha sacado CARA  
Ha ganado el jugador 1 en la tirada 2

EL RESULTADO HASTA EL MOMENTO ES:  
JUGADOR 1 --> 1 PUNTOS  
JUGADOR 2 --> 0 PUNTOS  
JUGADOR 3 --> 1 PUNTOS

El resultado de la tirada 3 es:  
El jugador 1 ha sacado CARA  
El jugador 2 ha sacado CRUZ  
El jugador 3 ha sacado CARA  
Ha ganado el jugador 2 en la tirada 3

EL RESULTADO HASTA EL MOMENTO ES:  
JUGADOR 1 --> 1 PUNTOS  
JUGADOR 2 --> 1 PUNTOS  
JUGADOR 3 --> 1 PUNTOS

El resultado de la tirada 4 es:  
El jugador 1 ha sacado CRUZ  
El jugador 2 ha sacado CRUZ  
El jugador 3 ha sacado CRUZ

Ha habido empate en la tirada 4  
EL RESULTADO HASTA EL MOMENTO ES:  
JUGADOR 1 --> 1 PUNTOS  
JUGADOR 2 --> 1 PUNTOS  
JUGADOR 3 --> 1 PUNTOS

El resultado de la tirada 5 es:  
El jugador 1 ha sacado CARA  
El jugador 2 ha sacado CARA  
El jugador 3 ha sacado CRUZ  
Ha ganado el jugador 3 en la tirada 5

EL RESULTADO HASTA EL MOMENTO ES:  
JUGADOR 1 --> 1 PUNTOS  
JUGADOR 2 --> 1 PUNTOS  
JUGADOR 3 --> 2 PUNTOS

El resultado de la tirada 6 es:  
El jugador 1 ha sacado CARA  
El jugador 2 ha sacado CARA  
El jugador 3 ha sacado CARA

Ha habido empate en la tirada 6

EL RESULTADO HASTA EL MOMENTO ES:

JUGADOR 1 --> 1 PUNTOS

JUGADOR 2 --> 1 PUNTOS

JUGADOR 3 --> 2 PUNTOS

El resultado de la tirada 7 es:

El jugador 1 ha sacado CARA

El jugador 2 ha sacado CRUZ

El jugador 3 ha sacado CRUZ

Ha ganado el jugador 1 en la tirada 7

EL RESULTADO HASTA EL MOMENTO ES:

JUGADOR 1 --> 2 PUNTOS

JUGADOR 2 --> 1 PUNTOS

JUGADOR 3 --> 2 PUNTOS

El resultado de la tirada 8 es:

El jugador 1 ha sacado CRUZ

El jugador 2 ha sacado CRUZ

El jugador 3 ha sacado CARA

Ha ganado el jugador 3 en la tirada 8

EL RESULTADO HASTA EL MOMENTO ES:

JUGADOR 1 --> 2 PUNTOS

JUGADOR 2 --> 1 PUNTOS

JUGADOR 3 --> 3 PUNTOS

El resultado de la tirada 9 es:

El jugador 1 ha sacado CARA

El jugador 2 ha sacado CARA

El jugador 3 ha sacado CRUZ

Ha ganado el jugador 3 en la tirada 9

EL RESULTADO HASTA EL MOMENTO ES:

JUGADOR 1 --> 2 PUNTOS

JUGADOR 2 --> 1 PUNTOS

JUGADOR 3 --> 4 PUNTOS

El resultado de la tirada 10 es:

El jugador 1 ha sacado CARA

El jugador 2 ha sacado CARA

El jugador 3 ha sacado CRUZ

Ha ganado el jugador 3 en la tirada 10

EL RESULTADO HASTA EL MOMENTO ES:

JUGADOR 1 --> 2 PUNTOS

JUGADOR 2 --> 1 PUNTOS

JUGADOR 3 --> 5 PUNTOS

EL GANADOR FINAL HA SIDO EL JUGADOR 3 CON 5 PUNTOS

**Ejercicio 7.** Escribir una clase que calcule el número de la suerte de una persona en base a su fecha de nacimiento, que deberá solicitarse por teclado. Para ello se sumará el día, mes y año de la fecha de nacimiento introducida y a continuación se sumarán las cifras del resultado de la suma inicial. La salida por pantalla debe ser:

Introduce el año de nacimiento (4 dígitos) y pulsa Enter

1995

Introduce el mes en nacimiento y pulsa Enter

12

Introduce el día de nacimiento y pulsa Enter

19

Tu número de la suerte es el 10

**Ejercicio 8.** Escribir un programa que genere e imprima un cuadrado latino de orden N. Un cuadrado latino de orden N es una matriz cuadrada en la que su primera fila contiene los N primeros números naturales y cada una de las siguientes es N-1 filas contiene la rotación de la fila anterior un lugar a la derecha. Como ejemplo de salida por pantalla, el cuadro latino de orden 4 es (usa espacios para separar cada columna):

1	2	3	4
4	1	2	3
3	4	1	2
2	3	4	1

**Ejercicio 9.** El operador ternario se puede ver como una forma compacta de utilizar un if-else para asignar el valor a una variable, dependiendo de si se cumple o no una condición. Su sintaxis es:

```
<variable> = <condición>?<valor1, si true>:<valor2, si false>
```

Por ejemplo, si numero es una variable de tipo entero declarada e inicializada:

```
String s = numero>=5?"aprobado":"suspenso"
```

Por lo tanto, para inicializar la variable s, se comprueba la condición, si es verdadera se le asigna el primer valor, si es falsa el segundo. Esta inicialización se puede hacer usando un if-else de la siguiente manera:

```
String s;  
if (numero>=5) {  
    s = "aprobado";  
} else {  
    s = "suspenso";  
}
```

Crear un programa que genere entre 10 y 50 números aleatorios en el rango entre 0 y 100. Debe guardar cuántos son pares y cuántos impares e imprimir ambos números al finalizar el programa. Usar el operador ternario para actualizar los valores de las variables en las que se guardan el número de pares e impares.

## Normas de entrega

Los ejercicios se deben **subir a Aula Global** hasta las 8:00 del Lunes 22 de Octubre de 2018. Se deberá subir un fichero comprimido **zip** con los archivos .java de la carpeta **src** (se deben subir únicamente esos archivos, no el proyecto Eclipse completo). El nombre del fichero será "s6-iniciales-del-alumno1-iniciales-del-alumno2.zip" (por ejemplo Lucía Pérez Gómez y Juan García Jiménez subirán un archivo llamado s6-lpg-jgj.zip). **Ambos miembros de la pareja deberán subir el fichero.** Se debe usar el enlace de entrega del grupo de laboratorio al que se pertenezca.