

Práctica 1: Estructuras secuenciales

- 1.- Realizar un programa que reciba un número como dato de entrada al programa, y a continuación muestre un mensaje mostrando un saludo e indicando cuál ha sido el número introducido.

Para poder pasar argumentos (datos) a un programa desde Eclipse, seleccione menú *Run*, opción *Run Configuration*, pestaña (x) = *Arguments*, y en el área de texto titulada “Program arguments”, teclearemos los distintos argumentos que queramos pasar a nuestro programa, separados por espacios en blanco, en el caso de cadenas de caracteres, se encerrarán entre comillas dobles.

Si queremos ejecutar el programa desde la consola de MS-DOS, teclearíamos en la consola lo siguiente:

```
java Ejercicio1 7
```

- 2.- Dado el siguiente programa, traza su ejecución y determina los resultados si los datos de entrada son: 7 y 11.

```
public static void main(String[] args) {  
    // Constantes  
    final int n = 10;  
  
    // Variables  
    int a, b, c=0;  
  
    a = Integer.parseInt(args[0]);  
    b = Integer.parseInt(args[1]);  
  
    c = a - b + n;  
    c = c + c;  
    b = a + b - c;  
    a = a + b - c;  
    a = n * a;  
    System.out.println(a);  
}
```

- 3.- El siguiente algoritmo convierte los días terrestres en años del planeta Júpiter. Puede que tenga algún error, si ese es el caso, corrígelo y haz que funcione.

```
public static void main(String[] args) {  
    int diasTerrestres;  
  
    diasTerrestres = Integer.parseInt(args[0]);  
  
    System.out.println("El número de años de Jupiter sería " +  
        (float)(diasTerrestres)/(float)(365*12));  
}
```

¿Para qué sirven los casting que se han puesto en la salida del resultado? Prueba a quitarlos y comprueba que ocurre.

- 4.- a) Observar el siguiente programa, que guarda en dos variables de tipo *byte*, las edades de dos personas, y en otras dos de tipo *short* su sueldo base.

```
class TiposEnteros1  
{  
    public static void main(String[] args) {  
        byte edadJuan = 20;  
        byte edadPedro = 42;  
        short sueldoBase = 1980;  
        short complementos = 400;  
        System.out.println(sueldoBase);  
    }  
}
```

b) Probar a cambiar los valores de inicialización, de forma que se pase del rango permitido, ¿qué ocurre?

c) Modificar el programa para que muestre por pantalla un mensaje similar al siguiente:

```
El sueldo real del empleado de XX años es: XXXX euros.  
El sueldo real del empleado de XX años es: XXXX euros.
```

Sabiendo que el sueldo real se obtiene de sumar al sueldo base los complementos, y 30 euros si su edad es inferior a 25, 60 euros si su edad es inferior a 35, y 100 euros si su edad es inferior o igual a 65, y después deducirle un 15% de IRPF y un 9% de Seguridad Social.

5.- Cuando se vaya a asignar a una variable de tipo *long* un valor grande habrá que ponerlo acabado en la letra L, por ejemplo 5000000000L. Esta letra indica que el valor debe ser tomado como *long* antes de ser asignado a la variable. Si no ponemos esta indicación, el valor numérico se toma como *int* (por defecto) y el compilador muestra un error indicando que el número es demasiado grande para pertenecer a ese tipo. Hacer la prueba con el siguiente ejemplo:

```
class TiposEnteros2  
{  
    public static void main(String[] args)  
    {  
        int habitantesEnMadrid = 4000000;  
        long habitantesEnElMundo = 5000000000L;  
        System.out.println(habitantesEnElMundo);  
    }  
}
```

6.- Al igual que ocurría con los *long*, puede ocurrir con los decimales (*float* y *double*). En el siguiente ejemplo establecemos el precio de una pieza de pan en 0.87 euros y el del Kilo de queso en 5.93 euros, la letra “f” con la que terminamos las declaraciones le indica al compilador que los literales numéricos (0.87 y 5.93) son de tipo *float*; si no pusiéramos esta indicación, el compilador los tomaría (por defecto) como *double*, con lo que la asignación *float = double* resultaría fallida. En el ejemplo estamos calculando el precio de un bocadillo de 150 gramos de queso. Hacer la prueba.

```
class TiposDecimales  
{  
    public static void main(String[] args)  
    {  
        float piezaPan = 0.87f;  
        float kiloQueso = 5.93f;  
        float bocadillo = piezaPan + (kiloQueso * 0.15f);  
        System.out.println("El bocadillo vale: " + bocadillo + " euros");  
    }  
}
```

Para definir e inicializar variables que contienen números realmente grandes (por ejemplo, 6 elevado a 100) y números realmente pequeños (por ejemplo, 2.45 elevado a -95), se puede utilizar el tipo *double* de la siguiente forma:

```
double var1 = 6E+100;  
double var2 = 2.45E-95;
```

No obstante, a pesar de lo explicado en los ejercicios 2 y 3, siempre se pueden realizar transformaciones tipo *Casting* (se verá en el apartado 11 del tema 4). Como avance veamos los siguientes ejemplos:

- (Byte) 1 → convierte el 1 (*int*) a *byte*.
- (double) miVariableDeTipoFloat → convierte a *double* una variable de tipo *float*.
- (short) (variableDeTipoByte + variableDeTipoByte) → convierte a *short* el resultado de sumar dos variables de tipo *byte*. Observar, que al sumar dos variables de tipo *byte*, el resultado puede que no “quepa” en otra variable de tipo *byte*.

- 7.- Realizar un programa que calcule la media aritmética de tres valores pasados como argumento a nuestro programa.
- 8.- Realizar un programa para calcular el valor de la hipotenusa de un triángulo rectángulo conocidos los dos catetos aplicando el teorema de Pitágoras.
- 9.- Escribir un programa que calcule el salario semanal de un trabajador, dada la tarifa horaria por horas y el número de horas trabajadas. Además hay que tener en cuenta que se le practica una retención del 4% para la S.S. y del 7% para el IRPF.
- 10.- Escribir un programa que convierta un número dado en segundos en el equivalente de minutos y segundos.
- 11.- Programa que recibe un número que corresponde al radio de una circunferencia, y calcula e imprime la longitud de la misma y el área del círculo correspondiente. ($longitud = 2 \cdot \pi \cdot r$) ($area = \pi \cdot r^2$)
- 12.- Realizar un programa que realiza y muestra las operaciones aritméticas básicas de dos números pasados a nuestro programa.
- 13.- Diseñar un algoritmo que a partir de la base y la altura de un triángulo da como resultado su superficie. ($superficie = b \cdot h / 2$).
- 14.- Un programa tiene como entrada dos números y los almacena en las variables “u” y “v”. ¿Qué hay que hacer par que al final del algoritmo los valores de las variables estén intercambiados? Diseña el algoritmo correspondiente.
- 15.- Escribir un programa que reciba 4 números naturales **a**, **b**, **c** y **d** y que calcule:

$$(a^b + c^d) * (a^c + b^d)$$
- 16.- Realizar un programa que reciba como dato de entrada el valor de una temperatura expresada en grados centígrados y nos calcule y escriba sus equivalentes en grados Reamhur, grados Farenheit y grados Kelvin.

$$100^\circ \text{ Centígrados} = 80^\circ \text{ Reamhur} = 212^\circ \text{ Farenheit} = 373^\circ \text{ Kelvin}$$

$$0^\circ \text{ Centígrados} = 0^\circ \text{ Reamhur} = 32^\circ \text{ Farenheit} = 273^\circ \text{ Kelvin}$$
- 17.- Diseñar un programa que, a partir de los tres puntos extremos (P1, P2 y P3) de un triángulo, calcule el área del mismo, de acuerdo con la siguiente fórmula:

$$Area = \sqrt{T * (T - S1) * (T - S2) * (T - S3)}$$

donde $T = S1 + S2 + S3 / 2$ y $S1, S2, S3$ son las longitudes de los lados del triángulo. La distancia entre dos puntos $p(x1, y1)$ y $q(x2, y2)$ es: $\sqrt{(x1 - x2)^2 + (y1 - y2)^2}$
- 18.- Convertir un tiempo expresado en segundo a un formato que lo represente a horas, minutos y segundos.
- 19.- Escribir una expresión en la que una variable numérica entera de nombre *cant* sea menor o igual que 500 y múltiplo de 5 pero distinto de 100.
- 20.- Escribir el resultado de la siguiente expresión:

$$35 > 47 \ \&\& \ 9 == 9 \ || \ 35 != 3 + 2 \ \&\& \ 3 >= 3$$