

Apéndice I

Boletines completos

BOLETÍN 1

Variables y condicionales

1. Pedir los coeficientes de una ecuación se 2º grado, y muestre sus soluciones reales. Si no existen, debe indicarlo.
2. Pedir el radio de un círculo y calcular su área. $A=PI*r^2$.
3. Pedir el radio de una circunferencia y calcular su longitud.
 $L=2*PI*r$.
4. Pedir dos números y decir si son iguales o no.
5. Pedir un número e indicar si es positivo o negativo.
6. Pedir dos números y decir si uno es múltiplo del otro.
7. Pedir dos números y decir cual es el mayor.
8. Pedir dos números y decir cual es el mayor o si son iguales.
9. Pedir dos números y mostrarlos ordenados de mayor a menor.

10. Pedir tres números y mostrarlos ordenados de mayor a menor.
11. Pedir un número entre 0 y 9.999 y decir cuantas cifras tiene.
12. Pedir un número entre 0 y 9.999 y mostrarlo con las cifras al revés.
13. Pedir un número entre 0 y 9.999, decir si es capicúa.
14. Pedir una nota de 0 a 10 y mostrarla de la forma: Insuficiente, Suficiente, Bien...
15. Pedir el día, mes y año de una fecha e indicar si la fecha es correcta. Suponiendo todos los meses de 30 días.
16. Pedir el día, mes y año de una fecha e indicar si la fecha es correcta. Con meses de 28, 30 y 31 días. Sin años bisiestos.
17. Pedir el día, mes y año de una fecha correcta y mostrar la fecha del día siguiente. suponer que todos los meses tienen 30 días.
18. Ídem que el ej. 17, suponiendo que cada mes tiene un número distinto de días (suponer que febrero tiene siempre 28 días).
19. Pedir dos fechas y mostrar el número de días que hay de diferencia. Suponiendo todos los meses de 30 días.
20. Pedir una hora de la forma hora, minutos y segundos, y mostrar la hora en el segundo siguiente.
21. Pedir una nota numérica entera entre 0 y 10, y mostrar dicha nota de la forma: cero, uno, dos, tres...
22. Pedir un número de 0 a 99 y mostrarlo escrito. Por ejemplo, para 56 mostrar: cincuenta y seis.

BOLETÍN 2
Condicionales y bucles

1. Leer un número y mostrar su cuadrado, repetir el proceso hasta que se introduzca un número negativo.
2. Leer un número e indicar si es positivo o negativo. El proceso se repetirá hasta que se introduzca un 0.
3. Leer números hasta que se introduzca un 0. Para cada uno indicar si es par o impar.
4. Pedir números hasta que se teclee uno negativo, y mostrar cuántos números se han introducido.
5. Realizar un juego para adivinar un número. Para ello pedir un número N, y luego ir pidiendo números indicando "mayor" o "menor" según sea mayor o menor con respecto a N. El proceso termina cuando el usuario acierta.
6. Pedir números hasta que se teclee un 0, mostrar la suma de todos los números introducidos.
7. Pedir números hasta que se introduzca uno negativo, y calcular la media.
8. Pedir un número N, y mostrar todos los números del 1 al N.
9. Escribir todos los números del 100 al 0 de 7 en 7.
10. Pedir 15 números y escribir la suma total.
11. Diseñar un programa que muestre el producto de los 10 primeros números impares.
12. Pedir un número y calcular su factorial.
13. Pedir 10 números. Mostrar la media de los números positivos, la media de los números negativos y la cantidad de ceros.
14. Pedir 10 sueldos. Mostrar su suma y cuantos hay mayores de 1000€.

15. Dadas las edades y alturas de 5 alumnos, mostrar la edad y la estatura media, la cantidad de alumnos mayores de 18 años, y la cantidad de alumnos que miden más de 1.75.
16. Pide un número (que debe estar entre 0 y 10) y mostrar la tabla de multiplicar de dicho número.
17. Una empresa que se dedica a la venta de desinfectantes necesita un programa para gestionar las facturas. En cada factura figura: el código del artículo, la cantidad vendida en litros y el precio por litro.
Se pide de 5 facturas introducidas: Facturación total, cantidad en litros vendidos del artículo 1 y cuantas facturas se emitieron de más de 600 €.
18. Igual que el anterior pero suponiendo que no se introduce el precio por litro. Solo existen tres productos con precios:
1- 0,6 €/litro, 2- 3 €/litro y 3- 1,25 €/litro.
19. Dadas 6 notas, escribir la cantidad de alumnos aprobados, condicionados (=4) y suspensos.
20. Pedir un número N, introducir N sueldos, y mostrar el sueldo máximo.
21. Pedir 10 números, y mostrar al final si se ha introducido alguno negativo.
22. Pedir 5 calificaciones de alumnos y decir al final si hay algún suspenso.
23. Pedir 5 números e indicar si alguno es múltiplo de 3.

BOLETÍN 3

Bucles anidados

1. Realiza detenidamente una traza al siguiente programa y muestra cual seria la salida por pantalla:

```
PROGRAMA ej_1
  VARIABLES
    suma, i, j: ENTERO
  COMIENZO
    PARA i <- 1 HASTA 4
      PARA j <- 3 HASTA 0 INC -1
        suma <- i*10+j
        escribir (suma)
      FIN PARA
    FIN PARA
  FIN
```

2. Realiza una traza del siguiente algoritmo y muestra la salida generada por pantalla.

```
PROGRAMA ej_1
  VARIABLES
    i, j: ENTERO
  COMIENZO
    PARA i <- 1 HASTA 3
      j <- i+1
      MIENTRAS j < 4
        escribir (j-i)
        j <- j+1
      FIN MIENTRAS
    FIN PARA
  FIN
```

3. Diseña una aplicación que muestre las tablas de multiplicar del 1 al 10.

4. Dibuja un cuadrado de n elementos de lado utilizando *.
5. Necesitamos mostrar un contador con 5 dígitos (X-X-X-X-X), que muestre los números del 0-0-0-0-0 al 9-9-9-9-9, con la particularidad que cada vez que aparezca un 3 lo sustituya por una E.
6. Realizar un programa que nos pida un número n, y nos diga cuantos números hay entre 1 y n que son primos.

BOLETÍN 4

Tablas

1. Leer 5 números y mostrarlos en el mismo orden introducido.
2. Leer 5 números y mostrarlos en orden inverso al introducido.
3. Leer 5 números por teclado y a continuación realizar la media de los números positivos, la media de los negativos y contar el número de ceros.
4. Leer 10 números enteros. Debemos mostrarlos en el siguiente orden: el primero, el último, el segundo, el penúltimo, el tercero, etc.
5. Leer por teclado dos tablas de 10 números enteros y mezclarlas en una tercera de la forma: el 1° de A, el 1° de B, el 2° de A, el 2° de B, etc.
6. Leer los datos correspondiente a dos tablas de 12 elementos numéricos, y mezclarlos en una tercera de la forma: 3 de la tabla A, 3 de la B, otros 3 de A, otros 3 de la B, etc.
7. Leer por teclado una serie de 10 números enteros. La aplicación debe indicarnos si los números están ordenados de forma creciente, decreciente, o si están desordenados.
8. Diseñar una aplicación que declare una tabla de 10 elementos enteros. Leer mediante el teclado 8 números. Después se debe pedir un número y una posición, insertarlo en la posición indicada, desplazando los que estén detrás.
9. Crear un programa que lea por teclado una tabla de 10 números enteros y la desplace una posición hacia abajo (el último pasa a ser el primero).

10. Ídem, desplazar N posiciones (N es introducido por el usuario).

11. Leer 5 elementos numéricos que se introducirán ordenados de forma creciente. Éstos los guardaremos en una tabla de tamaño 10. Leer un número N, e insertarlo en el lugar adecuado para que la tabla continúe ordenada.

12. Leer por teclado una tabla de 10 elementos numéricos enteros y leer una posición (entre 0 y 9). Eliminar el elemento situado en la posición dada sin dejar huecos.

13. Leer 10 enteros. Guardar en otra tabla los elementos pares de la primera, y a continuación los elementos impares.

Realizar dos versiones: una trabajando con los valores y otra trabajando con los índices.

14. Leer dos series de 10 enteros, que estarán ordenados crecientemente. Copiar (fusionar) las dos tablas en una tercera, de forma que sigan ordenados.

15. Leer 10 enteros ordenados crecientemente. Leer N y buscarlo en la tabla. Se debe mostrar la posición en que se encuentra. Si no está, indicarlo con un mensaje.

16. Queremos desarrollar una aplicación que nos ayude a gestionar las notas de un centro educativo. Cada grupo (o clase) está compuesto por 5 alumnos. Se pide leer las notas del primer, segundo y tercer trimestre de un grupo. Debemos mostrar al final: la nota media del grupo en cada trimestre, y la media del alumno que se encuentra en la posición N (N se lee por teclado).

BOLETÍN 5
Tablas n-dimensionales

1. Crear una tabla bidimensional de tamaño 5x5 y rellenarla de la siguiente forma: la posición T[n,m] debe contener n+m. Después se debe mostrar su contenido.
2. Crear y cargar una tabla de tamaño 4x4 y decir si es simétrica o no, es decir, si se obtiene la misma tabla al cambiar filas por columnas.
3. Crear y cargar dos matrices de tamaño 3x3, sumarlas y mostrar su suma.
4. Crear y cargar una tabla de tamaño 3x3, trasponerla y mostrarla.
5. Crear una tabla de tamaño 7x7 y rellenarla de forma que los elementos de la diagonal principal sean 1 y el resto 0.
6. Crear y cargar una tabla de tamaño 10x10, mostrar la suma de cada fila y de cada columna.
7. utilizando dos tablas de tamaño 5x9 y 9x5, cargar la primera y trasponerla en la segunda.
8. Crear una matriz "marco" de tamaño 8x6: todos sus elementos deben ser 0 salvo los de los bordes que deben ser 1. Mostrarla.
9. Hacer lo mismo que el ejercicio anterior, pero con una matriz 9x9x9. Es decir, creamos un cubo con las caras puestas a 1 y el interior a 0.
10. Los siguientes programas piden una serie de datos y tras procesarlos ofrecen unos resultados por pantalla. Mostrar el resultado:

```
PROGRAMA Ej10a
VARIABLES
  i, m, a: ENTEROS
  t: TABLA [5] ENTEROS
COMIENZO
  PARA i ← 0 HASTA 4
    leer (t[i])
```

```
PROGRAMA Ej10b
VARIABLES
  n, i: ENTEROS
  a, b: TABLA [100] ENTEROS
COMIENZO
  n ← 10
```

```

FIN PARA
m ← 0
PARA i ← 0 HASTA 4
    SI t[i] > m
        m ← t[i]
    FIN SI
FIN PARA
a ← t[4-m]
t[4-m] ← t[m]
t[m] ← a
PARA i ← 0 HASTA 4
    escribir (t[i])
FIN PARA
FIN PROGRAMA

```

Datos de entrada:

-4, 0, 1, 3 y 2.

```

PARA i ← 0 HASTA n-1
    leer (a[i])
FIN PARA
PARA i ← 0 HASTA n/2
    b[i] ← a[n-1-i]
    b[n-1-i] ← a[i]
FIN PARA
PARA i ← 0 HASTA n-1
    SI i mod 2 = 0
        escribir (a[i])
    SINO
        escribir (b[i])
    FIN SI
FIN PARA
FIN PROGRAMA

```

Datos de entrada:

6, 2, 8, 9, 2, 5, 8, 2, 6 y 1.

11-Se pretende realizar un programa para gestionar la lista de participaciones en una competición de salto de longitud. El número de plazas disponible es de 10. Sus datos se irán introduciendo en el mismo orden que vayan inscribiéndose los atletas. Diseñar el programa que muestre las siguientes opciones:

- 1- Inscribir un participante.
- 2- Mostrar listado de datos.
- 3- Mostrar listado por marcas.
- 4- Finalizar el programa.

Si se selecciona 1, se introducirán los datos de uno de los participantes: Nombre, mejor marca del 2002, mejor marca del 2001 y mejor marca del 2000.

Si se elige la opción 2, se debe mostrar un listado por número de dorsal.

La opción 3 mostrará un listado ordenado por la marca del 2002, de mayor a menor.

Tras procesar cada opción, se debe mostrar de nuevo el menú inicial, hasta que se seleccione la opción 4, que terminará el programa.

BOLETÍN 6
Funciones

1. Realizar una función, a la que se le pase como parámetro un número N, y muestre por pantalla N veces, el mensaje: "Módulo ejecutándose"
2. Diseñar una función que tenga como parámetros dos números, y que calcule el máximo.
3. Ídem una versión que calcule el máximo de 3 números.
4. Ídem una versión que calcule el máximo de una tabla de n elementos.
5. Función a la que se le pasan dos enteros y muestra todos los números comprendidos entre ellos, inclusive.
6. Función que muestra en pantalla el doble del valor que se le pasa como parámetro.
7. Realizar una función que calcule (muestre en pantalla) el área o el volumen de un cilindro, según se especifique. Para distinguir un caso de otro se le pasará el carácter 'a' (para área) o 'v' (para el volumen). Además hemos de pasarle a la función el radio y la altura.
8. Ídem que devuelva una tabla con el área y el volumen.
9. Módulo al que se le pasa un número entero y devuelve el número de divisores primos que tiene.
10. Ídem diseñar una función que devuelve una tabla con los divisores.
11. Escribir una función que calcule el máximo común divisor de dos números.
12. Ídem con tres números.

13. Ídem con una tabla.
14. Escribir una función que calcule el mínimo común múltiplo de dos números.
15. Ídem con tres números.
16. Ídem con una tabla.
17. Escriba una función que decida si dos números enteros positivos son amigos. Dos números son amigos, si la suma de sus divisores (distintos de ellos mismos) son iguales.
18. Diseña una función (en adelante **DUF**) que decida si un número es primo.
19. DUF que calcule a^n .
20. DUF que muestre en binario un número entre 0 y 255.
21. Escriba una función que sume los n primeros números impares.
22. Dado el valor de un ángulo, sería interesante saber su seno, coseno y tangente. Escribir una función que muestre en pantalla los datos anteriores.
23. Diseñar una función que calcule la distancia euclídea de dos puntos.
24. DUF a la que se le pasa como parámetro una tabla que debe rellenar. Se leerá por teclado una serie de números: guardaremos solo los pares e ignoraremos los impares. También hay que devolver la cantidad de impares ignorados.
25. DUF a la que se le pasa una tabla de enteros y un número. Debemos buscar el número en la tabla e indicar si se encuentra o no.

26. Igual que el ejercicio anterior, pero suponiendo que la tabla no está siempre llena, y el número de elementos se pasa también como parámetro.
27. Diseñar la función `opera_tabla`, a la que se le pasa dos tablas, el número de elementos útiles y que operación se desea realizar: sumar, restar, multiplicar o dividir (mediante un carácter: 's', 'r', 'm', 'd'). La función debe devolver una tabla con los resultados.
28. DUF que ordene la tabla que se le pasa.
29. DUF que toma como parámetros dos tablas. La primera con los 6 números de una apuesta de la primitiva, y la segunda con los 6 números ganadores. La función debe devolver el número de aciertos.
30. DUF recursiva que calcule a^n .
31. Calcular el factorial de n recursivamente.
32. DUF que calcule el valor máximo de una tabla de forma recursiva.
33. DUF que calcule el n -ésimo término de la serie de Fibonacci. En esta serie el n -ésimo valor se calcula sumando los dos valores anteriores. Es decir $\text{fibonacci}(n) = \text{fibonacci}(n-1) + \text{fibonacci}(n-2)$, siendo $\text{fibonacci}(0)=1$ y $\text{fibonacci}(1)=1$.
34. Igual que el ejercicio anterior, pero pudiendo configurar los valores de los dos primeros término de la serie.
35. DUF que realice la búsqueda dicotómica en una tabla, de forma recursiva.
36. DUF que toma una tabla bidimensional de enteros, representando un tablero de ajedrez. Disponemos de las constantes PB (peón blanco), TN (torre negra), etc. (P, T, C, A, R, D).

Dicho módulo debe devolver un valor booleano, que indique si el rey negro está amenazado.

37. Igual que el ejercicio anterior, pero indicando si existe jaque mate a las negras.

Apéndice II

Clase Entrada

```
import java.io.*;

public class Entrada {
    static String inicializar(){
        String buzon="";
        InputStreamReader flujo=new InputStreamReader(System.in);
        BufferedReader teclado=new BufferedReader(flujo);
        try{
            buzon=teclado.readLine();
        }
        catch(Exception e){
            System.out.append("Entrada incorrecta");
        }
        return buzon;
    }

    static int entero(){
        int valor=Integer.parseInt(inicializar());
        return valor;
    }

    static double real(){
        double valor=Double.parseDouble(inicializar());
```

```
        return valor;
    }

    static String cadena(){
        String valor=inicializar();
        return valor;
    }

    static char caracter(){
        String valor=inicializar();
        return valor.charAt(0);
    }
}
```