

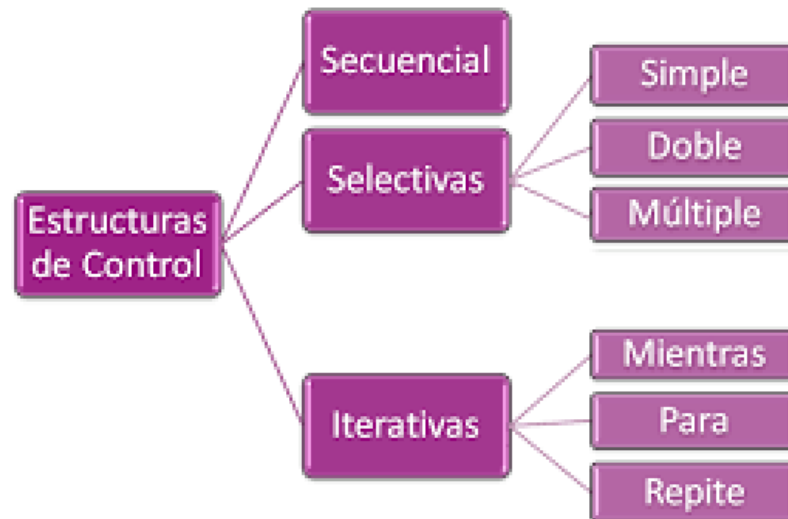
# DAM PROGRAMACIÓN



## UD2-USO DE ESTRUCTURAS DE CONTROL

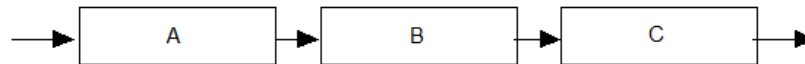
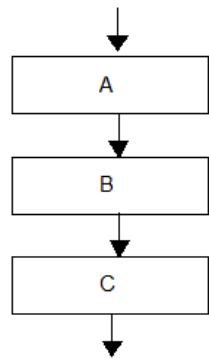
# Estructuras de control

- Las estructuras de control permiten modificar el flujo de ejecución de las instrucciones de un programa
- Todas las estructuras de control tienen un único punto de entrada.
- Las estructuras de control se pueden clasificar en: secuenciales, selectivas e iterativas.
- Básicamente lo que varía entre las estructuras de control de los diferentes lenguajes es su sintaxis, cada lenguaje tiene una sintaxis propia para expresar la estructura.



# Estructura secuencial

- Es una estructura con una entrada y una salida en la cual figuran una serie de acciones cuya ejecución es lineal y que se deben ejecutar en el orden en que aparecen.
- A su vez, todas las acciones tienen una única entrada y una única salida.



**Inicio**

**A  
B  
C**

**Fin**

```
/**
 * Estructura secuencial: Lee dos números por pantalla y muestra su suma
 */
public static void main(String[] args) {
    int n1, n2;
    Scanner miScanner = new Scanner(System.in);

    System.out.println("Introduce un número entero: ");
    n1 = miScanner.nextInt(); //lee un entero por teclado
    System.out.println("Introduce otro número entero: ");
    n2 = miScanner.nextInt(); //lee un entero por teclado

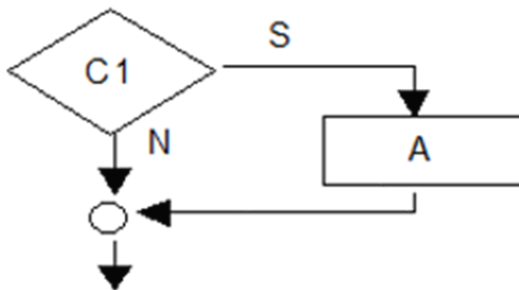
    System.out.println("Ha introducido los números: " + n1 + " y " + n2);
    miScanner.close();
}
```

```
/**
 * Muestra la hora y la fecha actual
 */
public static void main(String[] args) {
    Date miDate = new Date();
    SimpleDateFormat miDF1=new SimpleDateFormat ("dd-MM-yyyy");
    SimpleDateFormat miDF2=new SimpleDateFormat ("kk:mm:ss");

    System.out.println("La fecha actual es "+miDF1.format(miDate));
    System.out.println ("La hora actual es "+miDF2.format(miDate));
}
```

# Estructuras alternativas

- Una estructura alternativa es una estructura con una sola entrada y una sola salida, en la cual se realiza una acción entre varias, según una condición, o se realiza una acción según al cumplimiento o no de una determinada condición.
- En la condición de la estructura se puede usar
  - un valor booleano o
  - una expresión booleana que puede contener operadores condicionales (>, <, =, >=, etc) y lógicos (&&, ||)
- Las estructuras alternativas pueden ser:
  - **Simple**: solo se ejecuta alguna acción si la condición es cierta.
  - **Doble**: se ejecuta alguna acción si la condición es cierta y también si es falsa.
  - **Múltiple**: existen más de dos alternativas.

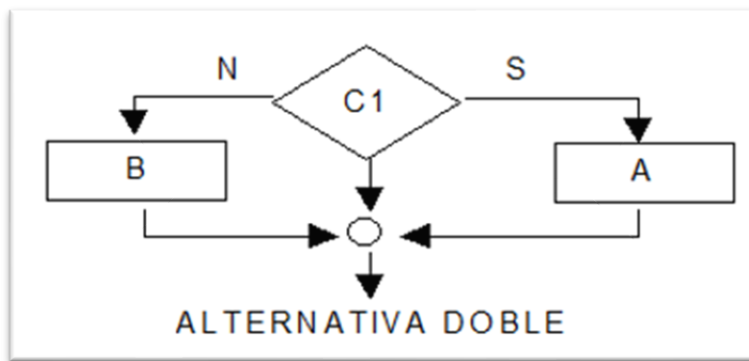


ALTERNATIVA SIMPLE

*Si C1*  
*A*  
*Fin si*

```
if(expresión_booleana){  
    instrucción 1  
    instrucción 2  
    .....  
}
```

```
/**  
 * Alternativa simple  
 * Indica si estás en el primer tramo de IRPF  
 */  
public static void main(String[] args) {  
    Scanner miScanner=new Scanner(System.in);  
    double ingresos  
  
    System.out.println("Introduce tus ingresos anuales");  
    ingresos=miScanner.nextDouble();  
  
    if (ingresos <12450){  
        System.out.println("No tienes que pagar IRPF");  
    }  
    miScanner.close();  
}
```



*Si C1  
entonces A  
sino B  
Fin si*

```
if(expresión booleana){  
    instrucciones 1  
}  
else{  
    instrucciones 2  
}
```

```
/**  
• Alternativa doble: Pregunta por la nota y si es >=5=>  
• aprobado sino ==>suspenso  
*/  
public static void main(String[] args) {  
    Scanner miScanner = new Scanner(System.in);  
  
    System.out.print("Introduce la nota del examen: ");  
    int nota = miScanner.nextInt();  
  
    if (nota >= 5) {  
        System.out.println("Enorabuena!! Has aprobado");  
    } else {  
        System.out.println("Lo Siento, has suspendido");  
    }  
    miScanner.close();  
}
```

```
/**
 * Estructuras de control condicionales dobles
 * Uso de cuadros de diálogo de la librería swing
 * Calcula la nota media de 3 notas dadas e informa al usuario
 * Es necesario importar la clase javax.swing.JOptionPane;
 */
```

```
public static void main(String[] args) {
    int notaM;
    int notaH;
    int notaI;
    double media;
```

```
//Uso de cuadros de diálogo de swing
```

```
notaM = Integer.parseInt(JOptionPane.showInputDialog("Nota de Matematicas"));
```

```
notaH = Integer.parseInt(JOptionPane.showInputDialog("Nota de Historia"));
```

```
notaI = Integer.parseInt(JOptionPane.showInputDialog("Nota de Informática"));
```

```
// Cálculo de la nota media
```

```
media = (notaMatematicas + notaHistoria + notaInformatica) / 3;
```

```
//Control de la nota media y uso de ventanas de swing
```

```
if (media >= 5) {
```

```
    JOptionPane.showMessageDialog(null, "¡Enhorabuena! Tu media es " + media);
```

```
} else {
```

```
    JOptionPane.showMessageDialog(null, "¡Recuperación! Tu media es " + media);
```

```
}
```

```
}
```

**If--else**

```
if(expresión booleana){
    instrucciones 1
}
else{
    instrucciones 2
}
```



## If anidados

```
if (expresion_booleana1) {  
    if (expresion_booleana2) {  
        instrucciones;  
    } else {  
        instrucciones;  
    }  
}
```

```
/**  
 * Estructura if anidada: Pregunta por el nombre, si es  
 * Ana pregunta por la  
 * edad y le dice si es mayor de edad o no  
 */  
public static void main(String[] args) {  
    int edad;  
    String nombre;  
  
    Scanner miScanner = new Scanner(System.in);  
    System.out.println("Introduce tu nombre");  
    nombre = miScanner.nextLine();  
    System.out.println("Introduce tu edad");  
    edad = miScanner.nextInt();  
  
    if (nombre.equals("Ana")) {  
        System.out.println("Hola Ana");  
        if (edad >= 18) {  
            System.out.println("Eres mayor de edad");  
        } else {  
            System.out.println("Aún no eres mayor de edad");  
        }  
    }  
    miScanner.close();  
}
```

```

/**
 * Estructura de control: if anidados
 * Genera un n° que representa la temperatura
 * Muestra un mensaje según el rango del n°
 */
public static void main(String[] args) {
    int temp;
    String mensaje;

    //Crear una objeto aleatorio
    Random elRandom = new Random();
    //Genera un n° entre 0 y 50
    temp = elRandom.nextInt(50);

    if(temp<10){
        mensaje = "Hace mucho frío.";
    }
    else{
        if(temp<15){
            mensaje = "Hace poco frío.";
        }
        else{
            if(temp<25){
                mensaje = "Hace una temperatura agradable.";
            }
            else{
                if(temp<30){
                    mensaje = "Hace poco calor.";
                }
                else{
                    mensaje = "Hace mucho calor.";
                }
            }
        }
    }

    System.out.printf("La temperatura actual es de %d grados.", temp);
    System.out.println("\n" + mensaje);
}

```

## If anidados

```

if (expresion_booleana1) {
    if (expresion_booleana2) {
        instrucciones;
    } else {
        instrucciones;
    }
}

```

```
/** Estructura de control: else if
 * Muestra la clasificación de la nota
 * en función de la nota numérica
 */
public static void main(String[] args) {
```

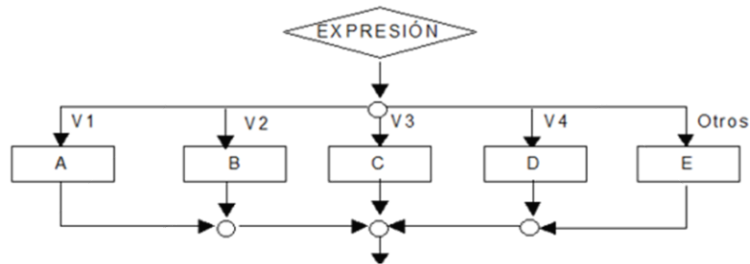
```
    Scanner miScanner = new Scanner(System.in);
    System.out.print("Nota(0 - 10): ");
    double nota = miScanner.nextDouble();
```

```
    if (nota == 10) {
        System.out.println("Matricula de Honor");
    } else if ((nota >= 9) && (nota < 10)) {
        System.out.println("Sobresaliente");
    } else if (nota >= 7) {
        System.out.println("Notable");
    } else if (nota >= 5) {
        System.out.println("Aprobado");
    } else {
        System.out.println("Suspenso");
    }
    miScanner.close();
}
```

```
if (expresion_booleana1) {
    instrucciones1
} else if (expresion_booleana2) {
    instrucciones2;
}
```

**else if**

## ALTERNATIVA MÚLTIPLE



*Opción EXPRESIÓN de*  
*V1 hacer A*  
*V2 hacer B*  
*V3 hacer C*  
*V4 hacer D*  
*V5 hacer E*  
*Fin opción*

**switch (expresión)**

```

{
  case valor1:
    declaracion1;
    break;
  case value2:
    declaracion2;
    break;
  .
  .
  case valorN:
    declaracionN;
    break;
  default:
    declaracionDefault;
}
  
```

```

/**
 * Estructura condicional switch. Pregunta por el tipo de coche
 * En función de valor introducido muestra información
 */
public static void main(String[] args) {
    String tipoVehiculo;

    Scanner miScanner = new Scanner(System.in);
    System.out.println("Introduce el tipo de coche");
    tipoVehiculo = miScanner.nextLine();

    switch (tipoVehiculo) {
        case "coche":
            System.out.println("Puedes pasar de 00:00 a 08:00");
            break;
        case "camion":
            System.out.println("Puedes pasar de 08:00 a 16:00");
            break;
        case "moto":
            System.out.println("Puedes pasar de 16:00 a 24:00");
            break;
        default:
            System.out.println("No puedes pasar con " + tipoVehiculo);
    }
}
  
```

**switch**  
 La expresión puede ser de tipo **byte**, **short**, **int**, **char**, **String** o una **enumeración**

```

/**Estructura condicional switch
 * Pregunta por el tipo de coche
 * En función de valor introducido muestra información
 */
public static void main(String[] args) {
    int dia;

    Scanner miScanner = new Scanner(System.in);
    System.out.println("Introduce el tipo de coche");
    dia = miScanner.nextInt();

    switch (dia) {
        case 2:
        case 3:
        case 4:
        case 5:
        case 6:
            System.out.println("Dia laboral");
            break;
        case 1:
        case 7:
            System.out.println("Fin de semana");
            break;
        default:
            System.out.println("La semana solo tiene 7
dias");
            break;
    }
    miScanner.close();
}

```

## switch

Los valores duplicados no están permitidos.

La opción **default** es opcional, pero si existe debe ser la última.

Cualquier declaración es opcional, si se omite, la ejecución continúa en el siguiente **case**.

# Ejercicios: estructuras condicionales

- Genera un organigrama de cada uno de los programas con la estructura **if/else** y **switch**.
- Implementa los siguientes algoritmos en Java:
  1. Determina si un número introducido por pantalla es par o no.
  2. Determina si un año es bisiesto o no (consulta en Internet las condiciones).
  3. Pedir el nombre de usuario y contraseña, mostrar un mensaje indicando si se corresponde con unos datos dados o no.
  4. Pedir 3 número y mostrarlos en orden de menor a mayor.
  5. Pedir un número entre 0 y 99999 y decir cuantas cifras tiene.
  6. Una farmacia desea un programa para ingresar el valor de compra y calcular lo siguiente: si el pago se efectúa al "contado", calcular un descuento del 5%; pero si el pago es con "tarjeta" se incrementa un recargo del 3% al valor de compra. Calcular y visualizar el descuento o recargo según sea el caso y el total a pagar de la compra.
  7. Hacer un programa que simule un cajero automático con un saldo inicial de 400€, con el siguiente menú de opciones:

`Elige el número de la operación:`

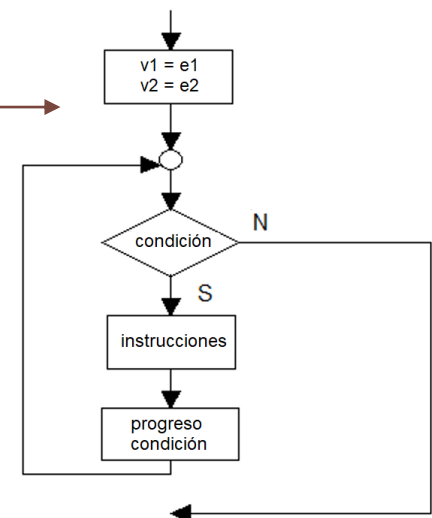
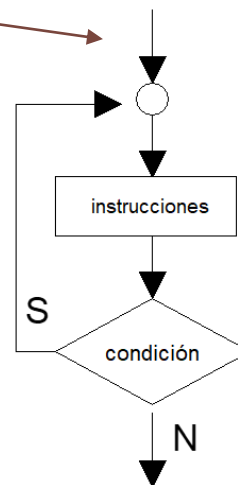
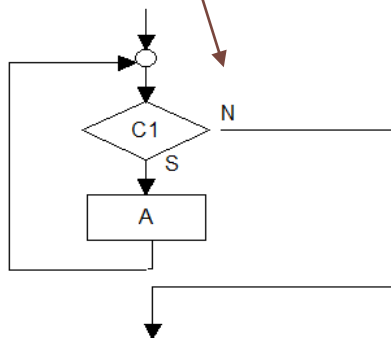
`1: Ingresar`

`2: Retirar`

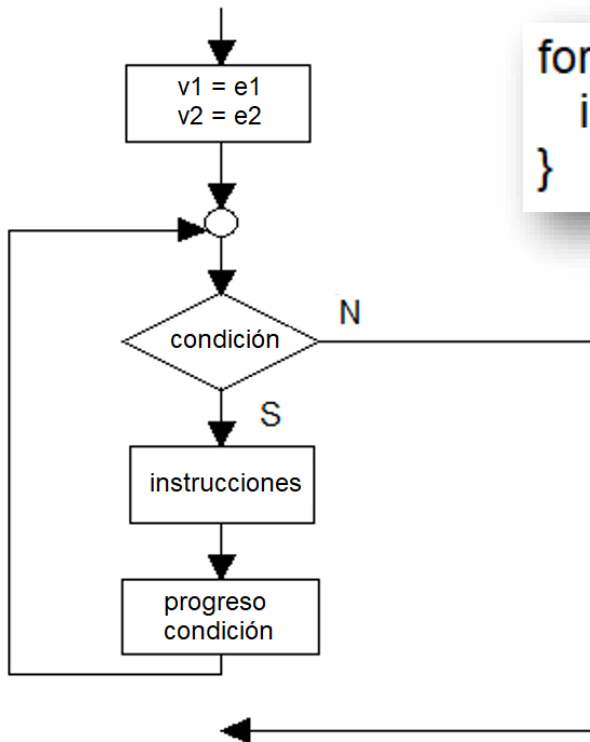
`· 3: Salir`

# Estructuras iterativas

- Es una estructura con una entrada y una salida en la cual se repite una acción un número de veces determinado o indeterminado de veces, dependiendo en este caso del cumplimiento de una condición.
- Las estructuras repetitivas pueden ser:
  - Estructura **for**
  - Estructura **while**
  - Estructura **do...while**



# Estructura for



```
for (v1 = e1, v2 = e2; condición; progreso-condición){  
    instrucciones  
}
```

*donde  $v1$  y  $v2$ , ... son variables de control que serán inicializadas con los valores  $e1$ ,  $e2$ , ...*

```
static void for_basico(){  
    for (int i=1;i<=10;i++){  
        System.out.println(i);  
    }  
}
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10



```
/** Estructura for: Muestra los n° pares
 * entre 0 y un n° leído por teclado*/
static void mostrarPares(){
    int num_maximo;
    Scanner miScanner = new Scanner(System.in);
    System.out.println("Introduce el n° máximo de "
        + "valores pares a mostrar");
    num_maximo = miScanner.nextInt();

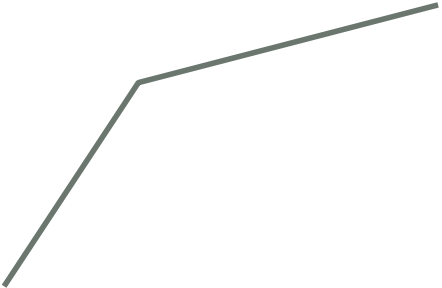
    for (int i=0; i<=num_maximo;i++){
        System.out.println(i);
    }
}
```

```
/*Muestra valores entre un nº mayor y menor
Introducidos por pantalla*/
static void mostrar_rango() {
    Scanner miScanner = new Scanner(System.in);
    System.out.println("Introduce el numero mayor");
    int mayor = miScanner.nextInt();

    System.out.println("Introduce el numero menor");
    int menor = miScanner.nextInt();

    if (menor > mayor) {
        System.out.println("Datos incorrectos");
    } else {
        //Voy decrementando desde el numero grande al pequeño
        for (int i = mayor; i >= menor; i--) {
            System.out.println("Estoy en el bucle");
            System.out.println(i);
        }
    }
    miScanner.close();
}
```

*Modifica el código: Pedir dos nº cualquiera y mostrar los valores entre uno y otro. No habrá nº incorrectos*



## ejemplos de bucles for infinitos

```
//for valores de inicio, ni condición ni incremento  
for( ; ; ){  
    System.out.println("Este es un ciclo infinito")  
}
```

```
//for con condición de salida que nunca se cumple  
for (int i=0; i<0; i++){  
    System.out.println("Nunca se cumple la condición");  
}
```

## bucles for anidados

1\*1 = 1  
2\*1 = 2  
3\*1 = 3  
4\*1 = 4  
5\*1 = 5  
6\*1 = 6  
7\*1 = 7  
8\*1 = 8  
9\*1 = 9

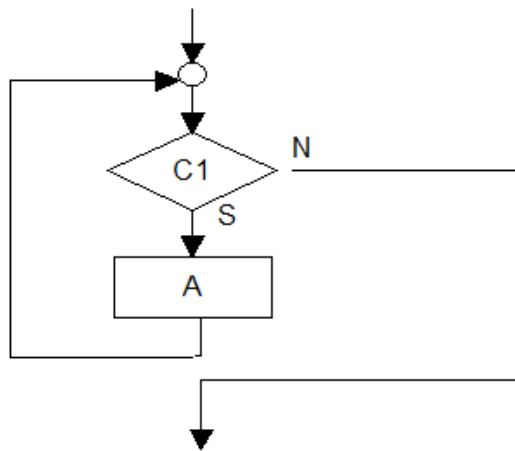
```
/**for anidados: Muestra la tabla de multiplicar
de los números del 1 al 9*/
static void mostrarTablaMultiplicar(){
    for (int i = 1; i <= 9; i++){
        //Para cada valor de i, se genera valores j
        //de 1 a 9
        for (int j = 1; j <= 9; j++){
            System.out.println(i + "*" + j + " = " + i*j);
        }
    }
}
```

# Ejercicios: bucle for

1. Calcular el factorial de un número que se pida por pantalla. El número debe estar entre 0 y 8.
2. Calcular el sumatorio de un número que se pida por pantalla.
3. Implementar un programa que pida un número entero mayor que 0 y nos muestre sus divisores.
4. Pedir 10 números. Mostrar la media de los números negativos y la cantidad de 0.
5. Comprobar si un número introducido por teclado es primo. Un número primo es aquel que solo es divisible entre sí mismo y entre 1. Un número es divisible entre otro si el resto es 0.
6. Crear un programa que dibuje una escalera de 10 números. Cada línea está formada por el nº1 y un número más que la anterior. La primera línea es un 1.

```
1
12
123
1234
12345
```

# Estructura while



`while (condición)`  
`sentencia;`

*las sentencias se ejecutan de forma repetida mientras se cumpla la condición*

/\*\*Ejemplo básico del funcionamiento del bucle while.

\* Muestra los número enteros del

\* Imprime por pantalla los números del 1 al 10\*/

```
static void while_basico() {
```

```
    int i=0;
```

← inicialización de la variable del bucle

```
    while (i<=10) {
```

← comprobación del estado de la variable del bucle

```
        System.out.println(i);
```

```
        i++;
```

← cambio de valor de la variable del bucle

```
    }
```

Si la condición no se cumple no se ejecuta el bucle

```
}
```

# Estructura while: Tablas de trazas

- Para comprender y utilizar correctamente los bucles es necesario conocer la evolución de la variable que controla la entrada y salida del bucle.
- Para facilitar esta labor se puede crear una tabla para hacer el seguimiento de los valores de las variables cada vez que se produzca una iteración del bucle.

```
int j=8;  
int i=0;  
  
while (i<=j) {  
    j/=2;  
    i++;  
}  
System.out.println(i+"-"+j);
```



nº iteración del bucle	j	i	(i<=j)
0(antes de entrar)	8	0	true
1	4	1	true
2	2	2	true
3	1	3	false
muestra por pantalla 3-1			

# Ejercicios: bucle while

1. Realiza un método que tenga el mismo resultado que el siguiente código:

```
for (int x = 10; x <= 100; x=x+10)
    System.out.println(x);
```

3. Corrige el siguiente código para que muestre los nº del 1 al 10.

```
int x=1;
while (x < 10)
{
    System.out.println(x);
}
```

4. Corrige el siguiente código para que muestre los números de 10 a 1

```
int x = 10;
while (x >= 0)
{
    x--;
    System.out.println(x);
}
```

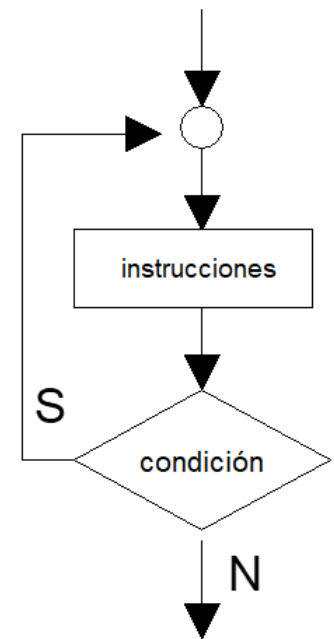
4. Calcular el factorial y el sumatorio de un número que se pida por pantalla (no mayor que 10) y mostrar al final los resultados.
5. Implementa un programa para jugar a adivinar un número que se genera de forma aleatoria entre 0 y 100.
  - Después de cada intento, el programa te dará pistas como "demasiado alto" o "demasiado bajo".
  - Debes permitir solo 10 intentos.



# Estructura do...while

*Las instrucciones se ejecutan hasta que se cumple la condición.  
La 1ª vez siempre se ejecuta*

```
do {  
    instrucciones  
}while (condición);
```



/\*\*Ejemplo básico del funcionamiento del bucle do...while. Muestra los números enteros del 1 al 10\*/

```
static void dowhile_basico() {
```

```
    int i=0;
```

← inicialización de la variable del bucle

```
    do{
```

```
        System.out.println(i);
```

```
        i++;
```

← cambio de valor de la variable del bucle

```
    }while (i<=10);
```

← comprobación del estado de la variable del bucle

```
}
```

El bucle siempre se ejecuta por lo menos 1 vez ya que la condición se analiza al final

# Ejercicios: bucle do..while

- Crea un programa muestre la tabla de multiplicar del 2 al 9.
- Crea un programa que pinte un rectángulo con \* en función de la altura y anchura que se pida por pantalla.

```
*****  
*               *  
*               *  
*****
```

- Realiza una mejora utilizando como carácter para el círculo un carácter ASCII gráfico. Utiliza en este caso el código hexadecimal para dar valor al carácter.
- Realiza el programa que simule a un cajero automático.
  - Debe pedir una contraseña para poder realizar las operaciones.
  - Debe mostrar un menú con las operaciones: ingresar dinero, retirar dinero, ver saldo, cambiar contraseña. Este menú debe utilizar do..while.
  - El saldo inicial se asigna desde el programa al inicio.

# Prácticas.

1. Escribe un programa que permita realizar el cifrado de César de un texto. Para realizar el cifrado César se necesita el texto a cifrar y un número que indica el desplazamiento que se emplea de los caracteres para cifrar. Si el número es 2, la letra 'a' se sustituye por la letra 'c', la 'b' por la 'd', etc. Utiliza para el cifrado el alfabeto español, que incluye la ñ.
2. Escribe un programa que pida solicite al usuario una cantidad de dinero en euros (puede tener decimales) para pagar una tableta de chocolate que vale 4, 65 €.El programa debe devolver las monedas o billetes según el dinero que entrega.
3. Se define la serie de Fibonacci como 1,1,2,3,5,8,13,21... donde los dos primeros términos son 1 y cada término restante es la suma de los dos anteriores. Formalmente, considerando el 0 como índice del primer elemento, cada termino  $a_i$  se calcula como:  
$$1 \text{ (si } i = 0, i = 1)$$
$$a_{i-1} + a_{i-2} \text{ (si } i > 1)$$
  - Solicitar al usuario el término **i-ésimo** que desea y mostrarlo por pantalla.