

INTRODUCCIÓN A LA PROGRAMACIÓN

**BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA
FACULTAD DE CIENCIAS DE LA COMPUTACIÓN**

Luz A. Sánchez Gálvez
sanchez.galvez@correo.buap.mx
luzsg@hotmail.com

Operadores Aritméticos

Operador Aritmético	Operación	Ejemplo	Resultado
\wedge	Potencia	4^3	64
*	Multiplicación	$8.25 * 7$	57.75
/	División	$15/4$	3.75
+	Suma	$125.78 + 62.50$	188.28
-	Resta	$65.30 - 32.33$	32.97
mod	Módulo (residuo)	$15 \text{ mod } 2$	1
div	División entera	$17 \text{ div } 3$	5

Operador	Jerarquía	Operación
\wedge	Mayor ↓ Menor	Potencia
*, /, mod, div		Multiplicación, División, Módulo, División entera
+, -		Suma, Resta



Operadores Relacionales

- ▶ Los operadores relacionales se utilizan para comparar expresiones.

Operador	Operación	Ejemplo	Resultado
==	Igual que	"Hola" == "Lola"	FALSO
!=	Diferente	'a' != 'b'	VERDADERO
<	Menor que	7 < 15	VERDADERO
>	Mayor que	22 > 11	VERDADERO
<=	Menor o igual que	35 <= 22	FALSO
>=	Mayor o igual que	35 >= 20	VERDADERO

Precedencia de Operadores

- ▶ Los operadores lógicos permitirán combinar dos o más *expresiones relacionales en una sola expresión, que se evalúa como verdadero(1) o falso (0).*

Operador Lógico	Jerarquía	Expresión Lógica	Significado
!	Mayor ↓ Menor	No P	No P No es cierto que P Es FALSO que P
&&		P y Q	P y Q $P \wedge Q$ P sin embargo Q
		P o Q	P o Q o P o Q o ambas Mínimo P o Q

Expresión

- ▶ Es una combinación de variables, constantes y operadores.
- ▶ Debe estar bien escrita, es decir, siguiendo las reglas de los operadores, análogas a las utilizadas en notación matemática.
- ▶ Cada expresión toma un valor que se determina tomando los valores de las variables y constantes implicadas y la ejecución de las operaciones indicadas.



Expresión

- ▶ Las **expresiones** pueden ser:
 - ▶ **Aritméticas:** Donde las variables y constantes son numéricas y las operaciones son aritméticas, una vez ejecutadas, su valor es numérico:
$$x + (b+5) + z * z;$$
$$z = PI * (y*x)/(w*w);$$
 - ▶ **Relacionales:** Utilizan signos de comparación (<, >, =, etc.) con valores de tipo numérico o carácter para producir un valor lógico (verdadero o falso), $(A-2) < (B-4)$
 - ▶ **Lógicas:** Combinan operadores lógicos (&&, ||, !) que operarán sobre valores lógicos, de acuerdo con sus tablas de verdad.
 $(A < B) || (A > B)$
 - ▶ **De Caracteres:** Que involucra cadenas de caracteres alfanuméricos.
- ▶ Unión de dos cadenas.

Ejemplo

- ▶ ¿Cuál es el valor de las variables A, B y AUX antes y después de cada paso?

1. $A = 10$;
2. $B = 13 + 7$;
3. $AUX = A + 15$;
4. $A = B$;
5. $B = A + 4$;
6. $AUX = AUX + 1$;

- ▶ Antes de la ejecución de las instrucciones, el valor de A, B, y AUX es indeterminado, es decir, el contenido de las posiciones de memoria correspondientes a ellas es desconocido.



Ejemplo

1. A toma el valor 10
2. La expresión $13+7$ se evalúa y B toma el valor resultante, 20
3. La expresión $A+15$ se evalúa con el valor de A (10), AUX toma el valor 25 (A no se ve afectada)
4. A toma el valor de B, es decir, 20 (B no cambia)
5. La expresión $A+4$ se evalúa con el valor de A (20); B toma el valor resultante 24 (A no se ve afectada)
6. La expresión $AUX+1$ se evalúa con el valor de AUX (25) ; AUX toma el valor resultante 26.



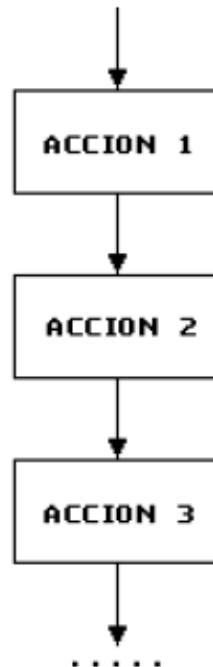
Estructuras de Control

- ▶ Las **estructuras de control** son las acciones que indican el orden de realización de los distintos pasos de un programa.
- ▶ Cada estructura tiene un punto de entrada y salida, lo que facilita la depuración de posibles errores.
- ▶ **Estructuras:**
 - ▶ **Secuenciales**
 - ▶ **Condicionales**
 - ▶ **Repetición**



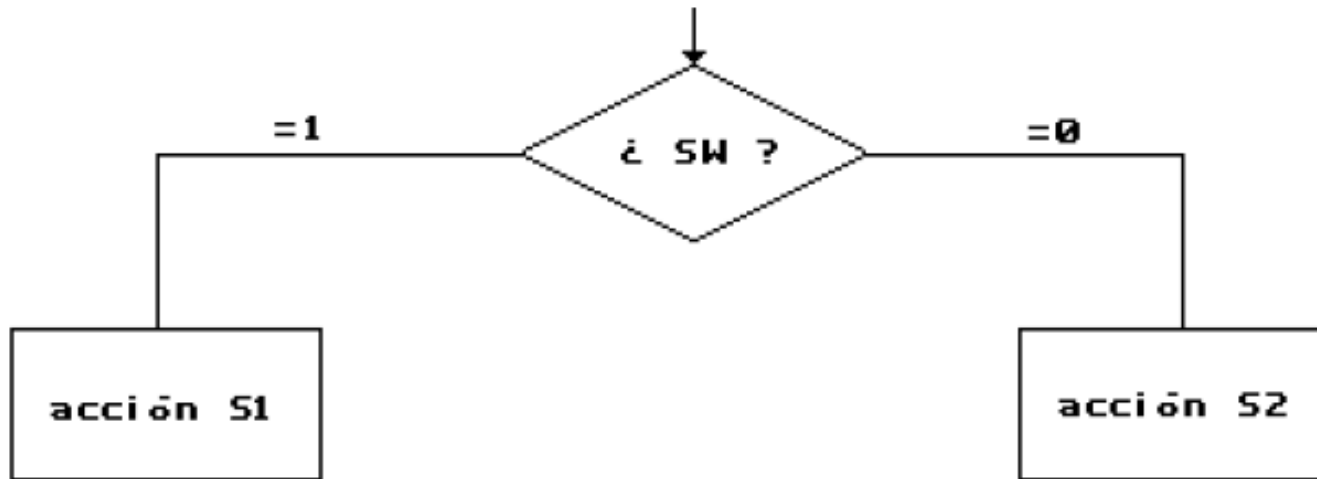
Estructuras Secuenciales

- ▶ Las **estructuras secuenciales** son aquellas en las que una acción (instrucción) sigue a otra de acuerdo con su orden de escritura.
- ▶ Las acciones suceden de tal modo que tras la salida (final) de una se efectúa la entrada (principio) de la siguiente y así sucesivamente hasta el fin del proceso.



Estructuras Condicionales

- ▶ Las **estructuras condicionales** se utilizan para tomar decisiones, de ahí que se suelen denominar también **estructuras de decisión** o alternativas.
- ▶ En estas estructuras se evalúa una **condición**, especificada mediante **expresiones lógicas** que en función del resultado, se realiza una opción u otra.



Estructuras Condicionales Simples

► Condicionales simples (si/if)

- La estructura condicional más sencilla, es la simple y se representa por **if**. Su efecto es el de ejecutar una determinada acción cuando se cumple una cierta condición y en caso contrario seguir el orden secuencial.
- La selección **if** evalúa la **condición**:
 - Si es **verdadera**, entonces ejecuta una o varias acciones .
 - Si es **falsa**, entonces no hace nada y sigue la ejecución normal del programa, pasando a la siguiente instrucción .



Estructuras Condicionales Simples

- ▶ La expresión:
 if (condición)
 <acción>

```
if (condición)  
{  
    <acciones>  
}
```



Estructuras Condicionales Simples

► Ejemplo.

- Suponga las variables $x=5$, $y = 100$

```
if (x == y)
```

```
    System.out.print("Las variables x e y tienen el mismo valor ");
```

- La condición se evalúa como falsa, por lo que no se imprimirá mensaje en pantalla.



Estructuras Condicionales Simples

► Ejemplo.

- Suponga las variables $x = 5$, $y = 5$

if ($x == y$)

`System.out.print("Las variables x e y tienen el mismo valor ");`

- La condición se evalúa como verdadera, por lo que en pantalla se visualizará el mensaje:

Las variables x e y tienen el mismo valor



Estructuras Condicionales Simples

► Ejemplo.

- Suponga las variables $x = 5$, $y = 5$

```
if (x == y)
```

```
{
```

```
    System.out.println("Las variables x e y tienen el mismo valor");
```

```
    System.out.println("X vale" x ) ;
```

```
    System.out.println("Y vale" y) ;
```

```
}
```

- La condición se evalúa como verdadera, por lo que en pantalla se visualizará el mensaje:

Las variables x e y tienen el mismo valor

X vale 5

Y vale 5



Estructuras Condicionales Simples

- ▶ Suponga las variables:
 - ▶ `COSTO_FINAL = 0.0, COSTO = 1989.50, INTERES = 50.00`
if (COSTO <= 5000 && INTERES == 50.00)
COSTO_FINAL = COSTO + INTERES ;
- ▶ La condición se evalúa como verdadera por lo que el valor actual de **COSTO_FINAL** es: **2039.50**



Estructuras Condicionales Simples

- ▶ Considere las clases Persona y Estudiante.
 - ▶ A la Clase Persona se le añade la variable de instancia **edad** y se renombra como **Persona1**.
 - ▶ A la Clase Estudiante se le añade la variable de instancia **promedio** y se renombra como **Estudiante1**.
 - ▶ Ahora se mostrará el uso de la **estructura condicional simple** en métodos y objetos.



Estructuras Condicionales Simples

- ▶ En la clase **Persona1** se añade el método `mayorEdad`, que consiste en determinar si se trata de una persona mayor de 18 años, al evaluar la condición como verdadera envía un mensaje indicando que la persona es mayor de edad.

```
public class Persona1{
```

```
...
```

```
    public void mayorEdad(){  
        if (edad >=18)
```

```
        System.out.println (nombre + " es una persona mayor de edad, pues  
        tiene " +edad +" años"); }  
    }
```

- ▶ Obviamente este método se debe invocar en el programa principal al crear un objeto de tipo `Persona` o `Estudiante`
 - ▶ `e.mayorEdad();`

Estructuras Condicionales Simples

- ▶ En el programa anterior, se compara el promedio de un estudiante. Como esta comparación se lleva a cabo en el programa principal se utiliza un objeto **e** que llama al método que muestra el promedio **getPromedio()**:

```
if (e.getPromedio() > 9)  
System.out.println (e.getNombre() + "es un alumno excelente");
```



Estructuras Condicionales Simples

► Comparación de Objetos

- Se crea el objeto p1 de la clase **Persona1** en el programa principal y se compara su edad con el objeto p.

```
public static void main(String[] args) {  
    ...  
    Persona p = new Persona("Daniel", "Puebla",18);  
    ...  
    Persona p1 = new Persona("Daniela", "Puebla",17);  
    if (p.getEdad() > p1.getEdad())  
        System.out.println (p.getNombre()+ " es mayor que " +p1.getNombre());  
}
```



Estructuras Condicionales Doble

- ▶ **Condicionales dobles (*si - sino / if - else*)**
 - ▶ Permite elegir entre dos opciones posibles en función del cumplimiento o no de una condición.
 - ▶ La selección ***if - else*** evalúa la condición:
 - ▶ Si es **verdadera**, se ejecuta la acción o acciones,
 - ▶ Si es **falsa**, se ejecuta la acción ó acciones, pasando en cualquier caso a la siguiente instrucción.
 - ▶ La sintaxis correspondiente es:
if (condición)
 <acciones>
else
 <acciones>

Estructuras Condicionales Doble

- ▶ Suponga que se tienen las variables $x = 5$, $y = 100$

if ($x == y$)

System.out.print("Las variables **x** e **y** tienen el mismo valor ");

else

System.out.print("La variables **x** tiene un valor diferente al
valor de la variable **y** ");

- ▶ Imprimirá en pantalla:

La variable **x** tiene un valor diferente al valor de la variable
y



Estructuras Condicionales Doble

- ▶ Suponga las variables NUM1= 50 y NUM2 =10, NUM3 = 0

if (NUM1 != 0)

NUM3 = NUM1 + NUM2

else

NUM3 = NUM1 – NUM2

- ▶ Después de llevarse a cabo “la decisión”, la variable **NUM3** tendrá el valor: **60**



Estructuras Condicionales Doble

- Suponga $k = 10$, $R = 2$;

```
if ( R < 100 )    // 2 < 100
{
    R = R + k;    //
    System.out.println(" R =" + R);
}
else
    System.out.println( " R = " + R);
```

- Imprimirá en pantalla: **R = 12**



Estructuras Condicionales Doble

- ▶ Suponga $k = 3$, $R = 110$

```
if ( R < 100 )    //110<100
{
    R = R + k;
    System.out.println( " R = " +R);
}
else
{
    R = R * k;    //110 *3
    System.out.println( " R = " + R);
}
```

- ▶ Imprimirá en pantalla: $R = 330$
-



Estructuras Condicionales Doble

```
Class Persona1{
```

```
...
```

```
void mayorEdad(){
```

```
    if (edad >= 18)
```

```
        System.out.print (nombre+" es un alumno mayor de edad, pues tiene "  
        +edad+" años" ) }
```

```
    else System.out.print (nombre +" es un alumno menor de edad, pues  
    tiene " +edad+ " años" )}
```

```
...
```

```
}
```

- ▶ **Obviamente este método se invoca desde el programa principal**

- ▶ e.mayorEdad();



Estructuras Condicionales Doble

- ▶ Se compara el promedio de un estudiante con el número 9, se accede al valor del promedio a través del objeto **e** que accede al método **getPromedio**.
- ▶ **Si el promedio es mayor que 9 se escribe:**
 - ▶ es un alumno excelente
- ▶ **Sino se escribe:**
 - ▶ es un buen alumno

...

```
if (e.getPromedio() > 9)
    System.out.print (e.getNombre() +" es un alumno excelente " );
else
    System.out.print (e.getNombre() +" es un buen alumno ");
```

...

Estructuras Condicionales Doble

- ▶ Anidada: en una condicional doble viene otra condicional

▶ ...

```
if (e.getPromedio() < 6)
```

```
    System.out.println (e.getNombre() +" es un alumno reprobado" );
```

```
else
```

```
    if (e.getPromedio() >= 6 && e.getPromedio() < 8 )
```

```
        System.out.println(e.getNombre() +" es un alumno regular ");
```

```
    else
```

```
        if (e.getPromedio() >= 8 && e.getPromedio() <= 9 )
```

```
            System.out.println (e.getNombre()+" es un buen alumno " );
```

```
        else System.out.println(e.getNombre()+" es un alumno excelente
```

```
        ");
```

```
    ...
```



Estructuras Condicionales Doble

► Comparación de Objetos

- Se crea el objeto p1 de la clase **Persona1** en el programa principal y se compara su edad con el objeto p.

Persona p1= new Persona ("Daniela", "Puebla",17);

if (p.getEdad() > p1.getEdad())

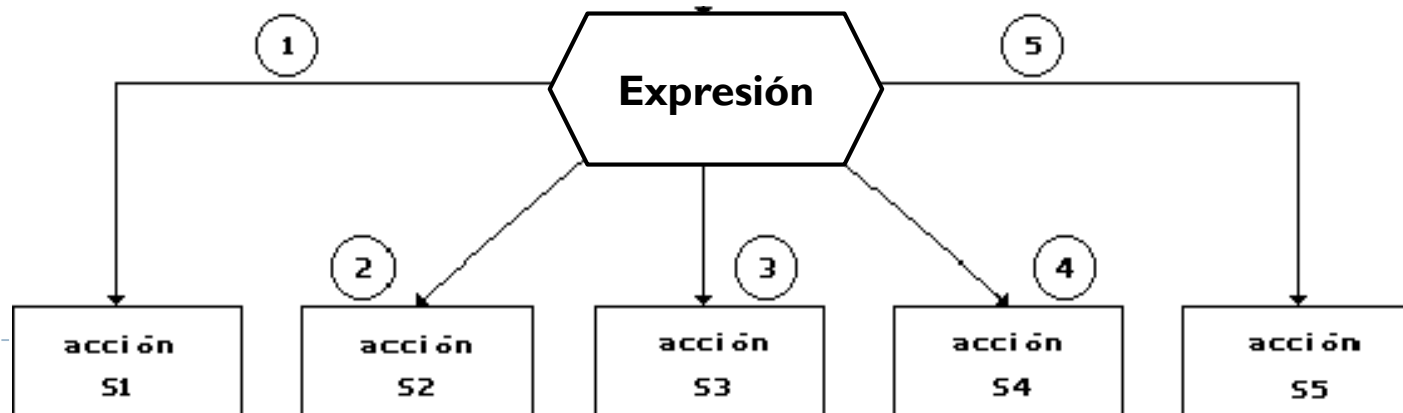
System.out.println (p.getNombre() +" es mayor que "
+p1.getNombre());

else System.out.println (p1.getNombre() + " es mayor
que " +p.getNombre());

Estructuras Condicionales Múltiple

► switch

- La estructura condicional múltiple evaluará una expresión que podrá tomar n valores (o grupos de valores) distintos e_1, e_2, \dots, e_n .
- Según el valor que tome la expresión, se realizará una de las n acciones posibles, es decir, el flujo del algoritmo seguirá un determinado camino entre los n posibles



Estructura Condicional Múltiple

- ▶ La estructura de decisión múltiple se puede representar como sigue, donde $\text{valor}_1, \dots, \text{valor}_n$ son los distintos valores (o grupos de ellos) que puede tomar la evaluación del selector que sirve de condición:

- ▶ **Sintaxis:**

```
switch ( selector )  
{  
    case  $\text{valor}_1$ : proposicion(es) 1;  
    break;  
    ....  
    case  $\text{valor}_n$ : proposicion(es) n;  
    break;  
    default: proposición x (esta proposición es opcional  
}
```



Estructura Condicional Múltiple

- ▶ **Selector** es una variable de tipo **entero o carácter**, que puede tomar solamente uno de los valores: $\text{valor}_1, \dots, \text{valor}_n$.
- ▶ Dependiendo del valor que tome selector, se ejecutará alguna de las proposiciones.
- ▶ Si selector toma valor_1 , entonces ejecutará la proposición 1, o bien si selector toma valor_2 , ejecutará la proposición 2, y así sucesivamente.
- ▶ Pero si selector **NO toma** alguno de los valores, valor 1 hasta valor n, entonces ejecutará la proposición por **default** (omisión) que es la proposición x.



Estructura Condicional Múltiple

```
{
...
switch ( e.getPromedio() )
{
    case 6: System.out.println (e.getNombre()+ " es un alumno en riesgo " );
        break;
    case 7: System.out.println (e.getNombre()+" es un alumno regular ");
        break;
    case 8: System.out.println (e.getNombre() + " es un buen alumno ");
        break;
    case 9: System.out.println (e.getNombre() + " es un alumno excelente ");
        break;
    case 10: System.out.println (e.getNombre() + " es un alumno magnifico ");
        break;
    default: System.out.println (e.getNombre() + " es un alumno reprobado ");
}
...
}
```



Estructura Condicional Múltiple

...

```
if (e.getPromedio() >=0 && e.getPromedio() <=10)
    switch ( e.getPromedio())
    {
        case 6: System.out.println (e.getNombre() + " es un alumno en riesgo ");
                break;
        case 7: System.out.println (e.getNombre() + " es un alumno regular " );
                break;
        case 8: System.out.println (e.getNombre() +" es un buen alumno ");
                break;
        case 9: System.out.println( e.getNombre() +" es un alumno excelente " );
                break;
        case 10: System.out.println (e.getNombre() +" es un alumno magnifico ");
                break;
        default: System.out.println (e.getNombre() + " es un alumno reprobado ");
    }
else System.out.println (" Promedio del alumno " + e.getNombre() +" fuera de
rango ");
}
```



...

```
public class ClaseClientePersona1 {
    public static void main(String[] args) {
        Persona p("Daniel", "Puebla", 18);
        p.imprimeDatos();
        p.mayorEdad();
        Persona p1("Daniela", "Puebla", 17);
        if (p.getEdad() > p1.getEdad())
            System.out.println (p.getNombre() + " es mayor que " + p1.getNombre());
        Estudiante e=new Estudiante ("Jacobo", "Merida", 19, "Ingenieria",9);
        e.imprimeDatos();
        System.out.println (e.getPromedio());
        e.mayorEdad();
        if (e.getPromedio() >=0 && e.getPromedio() <=10)
            switch ( e.getPromedio() )
            {
                case 6: System.out.println (e.getNombre() + " es un alumno en riesgo " );
                    break;
                case 7: System.out.println (e.getNombre() +" es un alumno regular “);
                    break;
                case 8: System.out.println (e.getNombre() +" es un buen alumno “);
                    break;
                case 9: System.out.println (e.getNombre() +" es un alumno excelente “);
                    break;
                case 10: System.out.println (e.getNombre() + " es un alumno magnifico “);
                    break;
                default: System.out.println (e.getNombre() +" es un alumno reprobado “);
            }
        else System.out.println ("Promedio del alumno " +e.getNombre() +" fuera de rango “);
    }
}
```

Estructura Condicional Múltiple

- ▶ Uno de los usos más frecuentes de una condicional múltiple es realizar un menú.
- ▶ De tal manera, que el usuario selecciona la opción que requiere llevar a cabo, facilitando la legibilidad del programa.



```

import javax.swing.JOptionPane;
public class ClaseClientePersonaI {
    public static void main(String[] args) {
        PersonaI p=new PersonaI ("Daniel", "Puebla",18);
        PersonaI pI=new PersonaI ("Daniela", "Puebla",17);
        EstudianteI eI=new EstudianteI ("Jacobo","Merida", 19,"Ingenieria",9);
        int opcion = Integer.parseInt(JOptionPane.showInputDialog("\t\t\t M E N U "+ "\n" +"\t 1: Mostrar los datos del objeto
Persona" + "\n" + "\t 2: Mostrar los datos del objeto Estudiante"+ "\n" +"\t 3: Comparar dos objetos "+" \n" +"\t 4: Enviar mensaje
de acuerdo a la calificación del alumno" ));
        switch(opcion)
        {
            case 1: System.out.println ("\n Los datos del objeto de la Clase Persona son ");
                p.imprimeDatos();
                p.mayorEdad();
                break;
            case 2: System.out.println( "\n Los datos del objeto de la Clase Estudiante son ");
                e.imprimeDatos();
                e.mayorEdad();
                break;
            case 3: System.out.println ( "\n Comparación de dos Objetos ");
                if (p.getEdad() > pI.getEdad())
                    System.out.println (p.getNombre()+" es mayor que "+pI.getNombre());
                break;
            case 4: System.out.println ("\n Comparación Calificación ");
                if (e.getPromedio() >=0 && e.getPromedio() <=10)
                    switch (e.getPromedio())
                    {
                        case 6: System.out.println ( e.getNombre() +" es un alumno en riesgo " ); break;
                        case 7: System.out.println ( e.getNombre() + " es un alumno regular " ); break;
                        case 8: System.out.println ( e.getNombre() + " es un buen alumno " ); break;
                        case 9: System.out.println ( e.getNombre() + " es un alumno excelente " ); break;
                        case 10: System.out.println ( e.getNombre() + " es un alumno magnifico " ); break;
                        default: System.out.println ( e.getNombre() + " es un alumno reprobado " );
                    }
                else System.out.println (" Promedio del alumno " + e.getNombre() + " fuera de rango ");}}}

```