

```
mirror_mod = modifier_ob.  
set mirror object to mirror.  
mirror_mod.mirror_object =  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True
```

```
selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier  
mirror_ob.select = 0  
= bpy.context.selected_obj  
data.objects[one.name].se  
print("please select exactly
```

```
--- OPERATOR CLASSES ---
```

```
types.Operator):  
X mirror to the selected  
object.mirror_mirror_x"  
mirror X"
```

```
context):  
context.active_object is not
```

Estructuras de decisión

ING. CARLOS H. RUEDA C.



CONTENIDO

1. Estructuras de decisión
 - 1.1. Estructuras de decisión simples
 - 1.2. Estructuras de decisión dobles
 - 1.3. Estructuras de decisión múltiple
2. Estructura múltiple
3. Ejemplos
4. Ejercicios

1. Estructuras de decisión

Son utilizadas en los algoritmos para tomar decisiones: si ocurre cierta condición, se ejecutan ciertas instrucciones; en caso contrario, se ejecutan otras.

1. Estructuras de decisión

Toda decisión implica una evaluación, generalmente una comparación entre variables o contra un valor específico.

1. Estructuras de decisión

Dependiendo del resultado de esta comparación, el programa sigue un curso de acción particular.

1. Estructuras de decisión

Pueden ser:

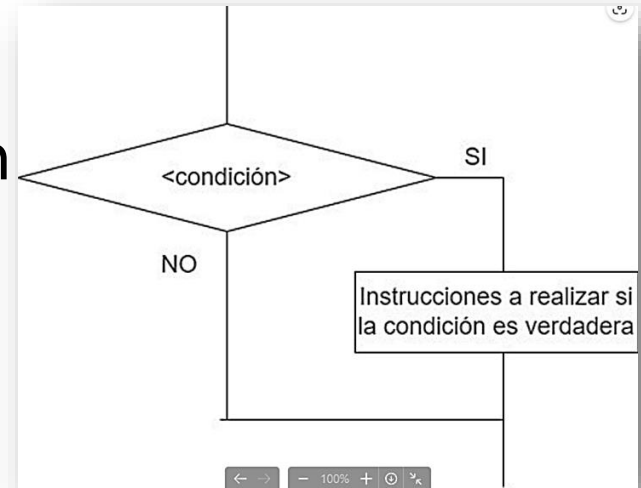
- ✓ Simples,
- ✓ Dobles
- ✓ Múltiples

1.1. Estructuras de decisión simples

se activan solo cuando la expresión evaluada produce un resultado afirmativo. Su estructura típica es:

1.1. Estructuras de decisión simples

se activan solo cuando la expresión evaluada produce un resultado afirmativo. Su estructura típica es:



1.1. Estructuras de decisión simples

se activan solo cuando la expresión evaluada produce un resultado afirmativo. Su estructura típica es:

```
Si <condición> entonces  
    Instrucción (es)  
Fin-Si
```

1.2. Estructuras de decisión dobles

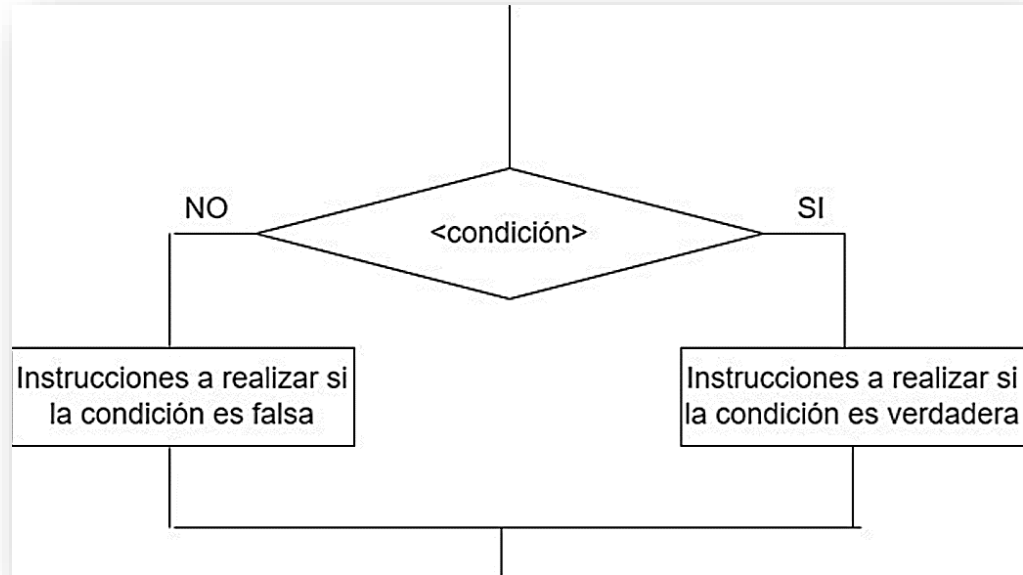
Ofrecen la posibilidad de seleccionar entre dos alternativas en base al cumplimiento de una condición específica.

1.2. Estructuras de decisión dobles

Si la expresión evaluada es verdadera, el programa tomará un camino, y si es falsa, seguirá otro.

1.2. Estructuras de decisión dobles

Se visualiza:



1.2. Estructuras de decisión dobles

Se visualiza:

```
Si <condición> entonces  
    Instrucción (es)  
Si no  
    Instrucción (es)  
Fin-Si
```

1.3. Estructuras de decisión múltiples

Son decisiones especializadas que involucran la comparación de una variable con varios resultados posibles.

1.3. Estructuras de decisión múltiples

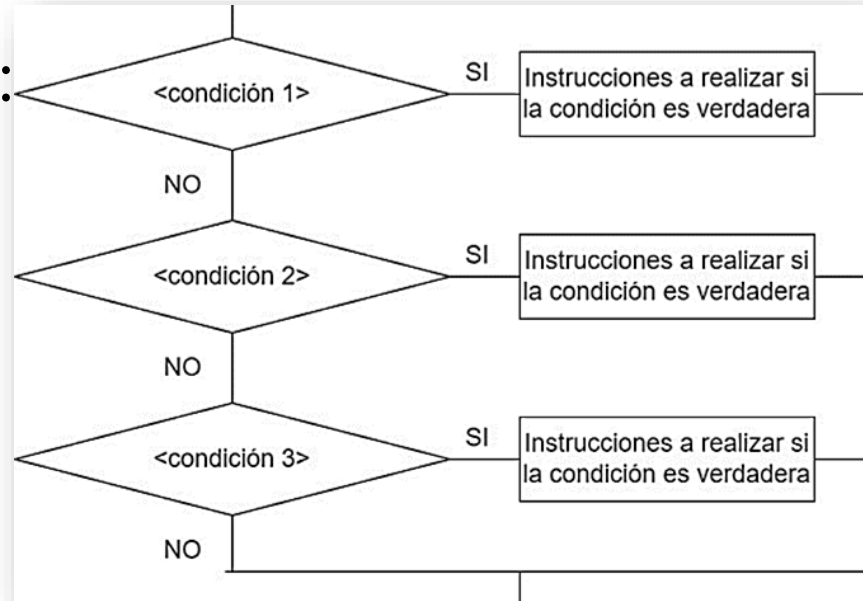
Para cada resultado, se ejecutan instrucciones específicas.

1.3. Estructuras de decisión múltiples

No se limita a uno o dos caminos; se pueden tener tantos caminos como se necesiten en un algoritmo.

1.3. Estructuras de decisión múltiples

La estructura típica es:



1.3. Estructuras de decisión múltiples

La estructura típica es:

```
Si <condición> entonces
    Instrucción (es)
Si no
    Si <condición> entonces
        Instrucción (es)
    Si no
        .
        .
        .
Fin-Si
```

2. Estructura múltiple

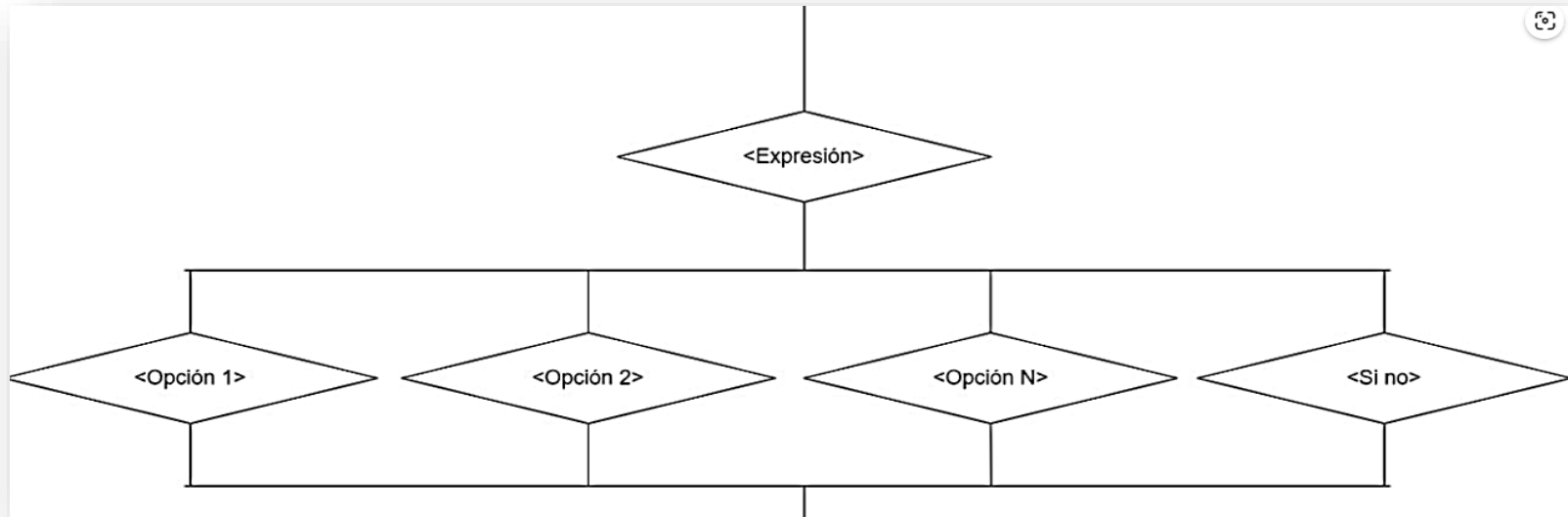
Las estructuras de comparación múltiples, mediante la cláusula "en caso de", posibilitan la comparación de una variable o constante con varios valores potenciales.

2. Estructura múltiple

Esta estructura representa una decisión especializada que permite establecer múltiples rutas dentro de un algoritmo, dependiendo de los diferentes valores posibles de la variable.

2. Estructura múltiple

Su estructura básica es la



2. Estructura múltiple

Su estructura básica es la

```
En-caso-de <expresión> haga
  Caso <opción 1>:
    <Instrucciones>
  Caso <opción 2>:
    <Instrucciones>
  Caso <opción 3>:
    <Instrucciones>
  ...
  Caso <opción N>:
    <Instrucciones>
  Si-no
    <Instrucciones a realizar si no se ha cumplido ninguna de las condiciones anteriores>
Fin-Caso
```

3. Ejemplos

Estructuras de Decisión Sencillas (si)

Escribe un programa que solicite al usuario ingresar un número entero. El programa debe imprimir en pantalla si el número es positivo.

3. Ejemplos

Estructuras de Decisión Sencillas (si)

Crea un programa que pida al usuario su edad y verifique si es mayor de 18 años. Si lo es, debe mostrar el mensaje: "Eres mayor de edad".

3. Ejercicio

Estructuras de Decisión Sencillas (si)

Desarrolla un programa que reciba una temperatura en grados Celsius y verifique si está por encima de 30 grados. Si es así, debe mostrar el mensaje: "Hace calor hoy".

3. Ejemplo

Estructuras de Decisión Dobles (si-no)

Escribe un programa que solicite al usuario ingresar un número entero y determine si el número es par o impar.

3. Ejemplo

Estructuras de Decisión Dobles (si-no)

Crea un programa que pida al usuario ingresar una contraseña. Si la contraseña es correcta (elige una contraseña predeterminada), muestra el mensaje "Acceso concedido", de lo contrario, muestra "Acceso denegado".

3. Ejercicio

Estructuras de Decisión Dobles (si-no)

Desarrolla un programa que solicite al usuario ingresar una nota entre 0 y 100. Si la nota es mayor o igual a 60, el programa debe imprimir "Aprobado"; si no, debe imprimir "Reprobado".

3. Ejemplo

Estructuras de Decisión Múltiples (si-no-si)

Escribe un programa que solicite al usuario ingresar un número entero y determine si el número es positivo, negativo o cero.

3. Ejemplo

Estructuras de Decisión Múltiples (si-no-si)

Desarrolla un programa que solicite al usuario ingresar la **cantidad de horas trabajadas** en una semana y calcule el **pago** correspondiente:

- Hasta 40 horas: se paga la **tarifa normal** (por ejemplo, \$10 por hora).
- De 41 a 50 horas: se paga la **tarifa normal** por las primeras 40 horas y **1.5 veces la tarifa** por las horas adicionales.
- Más de 50 horas: se paga la **tarifa normal** por las primeras 40 horas, **1.5 veces la tarifa** por las siguientes 10 horas y **doble tarifa** por las horas que excedan de 50.

3. Ejercicio

Estructuras de Decisión Múltiples (si-no-si)

Crea un programa que pida al usuario ingresar una **calificación** (entre 0 y 100) y muestre la letra correspondiente:

- ✓ 90 o más: **A**
- ✓ 80 a 89: **B**
- ✓ 70 a 79: **C**
- ✓ 60 a 69: **D**
- ✓ Menos de 60: **F**

3. Ejemplo

Estructura Múltiple (caso)

Escribe un programa que solicite al usuario ingresar un **número del 1 al 7** y muestre el **día de la semana** correspondiente. Por ejemplo, 1 corresponde a "Lunes", 2 a "Martes", etc.

3. Ejemplo

Estructura Múltiple (caso)

Crea un programa que pida al usuario seleccionar una **opción de menú** ingresando un número:

- **1:** Crear archivo
- **2:** Editar archivo
- **3:** Eliminar archivo
- **4:** Salir El programa debe ejecutar una acción o mostrar un mensaje correspondiente a la opción seleccionada.

3. Ejercicio

Estructura Múltiple (caso)

Desarrolla un programa que solicite al usuario ingresar una **letra que represente una nota** (A, B, C, D, F) y muestre el **comentario** correspondiente:

- ✓ **A:** Excelente
- ✓ **B:** Bueno
- ✓ **C:** Regular
- ✓ **D:** Malo
- ✓ **F:** Reprobado Si ingresa una letra diferente, debe mostrar "Nota no válida"

4. Ejercicios

Su estructura básica es la siguiente: