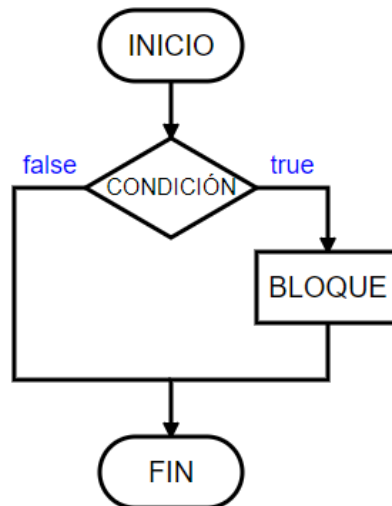


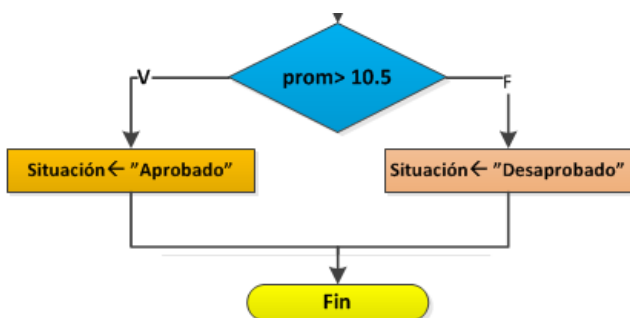
U5. USO DE ESTRUCTURAS DE CONTROL.

1. ESTRUCTURA DE SELECCIÓN.

a. SENTENCIA IF.



b. SENTENCIA IF-ELSE.



Condición

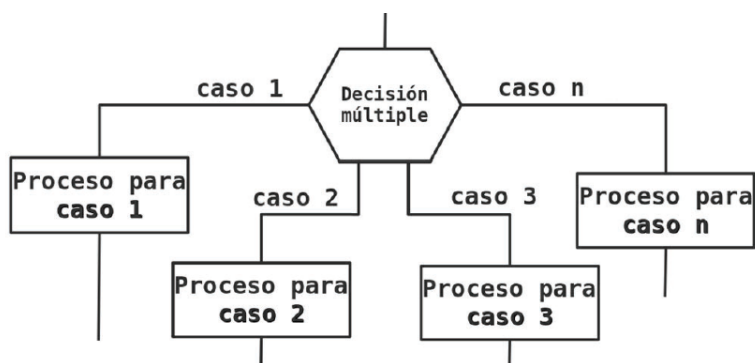
```
if (promedio >= 10.5) {  
    situacion = "Aprobado";  
}  
else {  
    situacion = "Desaprobado";  
}
```

Estructura Condicional

Instrucciones si es Verdadera la Condición

Instrucciones si es Falsa la Condición

c. SENTENCIA SWITCH.



```
String mensaje;  
int valorEntero;  
switch(valorEntero) {  
    case 1: mensaje = "Uno";  
        break;  
    case 2: mensaje = "Dos";  
        break;  
    case 3: mensaje = "Tres";  
        break;  
    default: mensaje = "desconocido";  
}
```

expresión entera que se evalúa

Se ejecutan las sentencias asociadas al case que cumple la condición

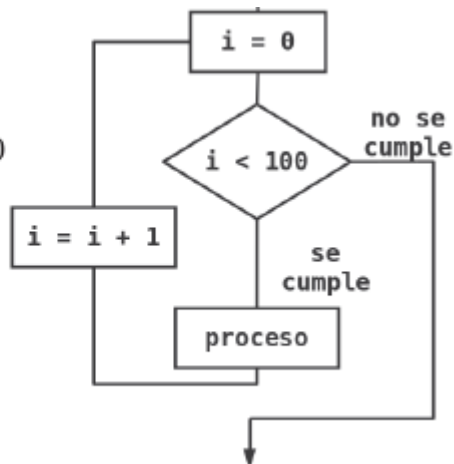
Finaliza el bloque

Si el valor no coincide con ninguno de los listados en las etiquetas case, se ejecuta la parte asociada a default (opcional)

2. ESTRUCTURA DE REPETICIÓN.

a. WHILE

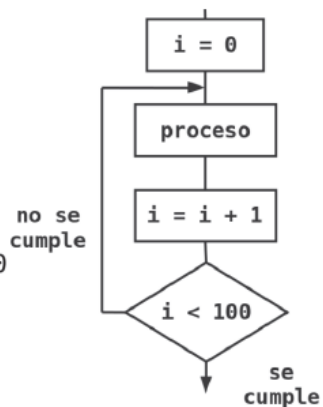
```
inicio
.....
i = 0
mientras i < 100
  Proceso
  i = i + 1
fin mientras
.....
fin
```



```
while ( expresion_booleana ) {
  sentencias;
}
```

b. UNTIL

```
inicio
.....
i = 0
hacer
  Proceso
  i = i + 1
mientras i < 100
.....
fin
```



```
do {
  sentencias;
} while( expresion_booleana )
```

c. FOR

```
for(int i = valor inicial; i <= valor final; i = i + paso)
{
  ....
  ....
  Bloque de Instrucciones....
  ....
  ....
}
```

```
for (int i=0; i<100; i++) {
  System.out.println(i);
}
```

3. ESTRUCTURAS DE SALTO: ETIQUETAS Y FUNCIONES.

a. ~~ETIQUETA CONTINUE Y BREAK.~~

Permite redirigir el flujo del programa.

Uso: En estructuras de repetición permite interrumpir la ejecución de estos.

~~a.1. break [etiqueta]~~

Permite salir de un bucle de repetición o un "Switch".

En caso de no indicar etiqueta saldrá de la secuencia de control del nivel actual.

```
uno: for(.....) {  
    dos: for(.....) {  
        tres: for(.....) {  
            break;           // saldría del bucle tres  
        }  
        break;           // saldría del bucle dos  
    }  
    break;           // saldría del bucle uno  
}
```

Si la sentencia break incluye etiqueta es posible saltar más de un nivel.

~~a.2. continue [etiqueta]~~

Finaliza el cuerpo del bucle yendo directamente a la evaluación de la misma. Se usa para saltar alguna instrucción dentro de un bucle de repetición

```
uno: for( ) {  
    dos: for( ) {  
        .....  
        continue;           // seguiría en el bucle interno  
        .....  
        continue uno; // seguiría en el bucle externo  
        .....  
        break;           // saldría del bucle interno  
        .....  
        break uno;       // saldría del bucle externo  
    }  
}
```

b. FUNCIONES Y PROCEDIMIENTO.

b.1 PROCEDIMIENTO: Conjunto de instrucciones que no retornan ningún valor. (void).

```
Algoritmo Procedimientos
  Escribir "Tabla de multiplicación"
  Escribir "Ingrese un número:"
  Leer input
  tablaMultiplicacion(input)
FinAlgoritmo

Subproceso tablaMultiplicacion(value)
  Escribir "Procesando tabla de multiplicacion..."
  Para index<-0 Hasta 10 Con Paso 1 Hacer
    calculo <- (value * index)
    Escribir index, ":", calculo
  Fin Para
FinSubproceso
```

```
public class Procedimiento_Funciones {

    public static void main(String[] args) {

        Scanner entrada_escaner = new Scanner(System.in);

        System.out.println("Tabla de multiplicación");
        System.out.println("Ingrese un número");
        int input = entrada_escaner.nextInt();
        tablaMultiplicacion(input);
    }

    public static void tablaMultiplicacion(int value) {
        System.out.println("Procesando tabla de multi..");
        for (int i = 0; i < 11; i++) {
            int calculo = value * i;
            System.out.println(value+"x"+i+"="+calculo);
        }
    }
}
```

b.2. FUNCIÓN: Conjunto de instrucciones que retornan un valor. (return).

```
Funcion resultado <- Maximo( n1, n2)
  Si n1 > n2 Entonces
    resultado <- n1
  Sino
    resultado <- n2
  FinSi
FinFuncion

Algoritmo Funciones01
  Escribir "El maximo de 5 y 12 es:"
  Escribir Maximo(5,12)
  Escribir "El maximo de 25 y 12 es:"
  Escribir Maximo(25,12)
FinAlgoritmo
```

```
public class Procedimiento_Funciones {

    public static void main(String[] args) {

        System.out.println("El máximo de 25 y 12 es: " + Maximo(5,12));
        System.out.println("El máximo de 25 y 12 es: " + Maximo(25,12));
    }

    public static int Maximo(int n1, int n2) {
        System.out.println("Funcion maximo");
        if (n1>n2) {
            return n1;
        }else {
            return n2;
        }
    }
}
```