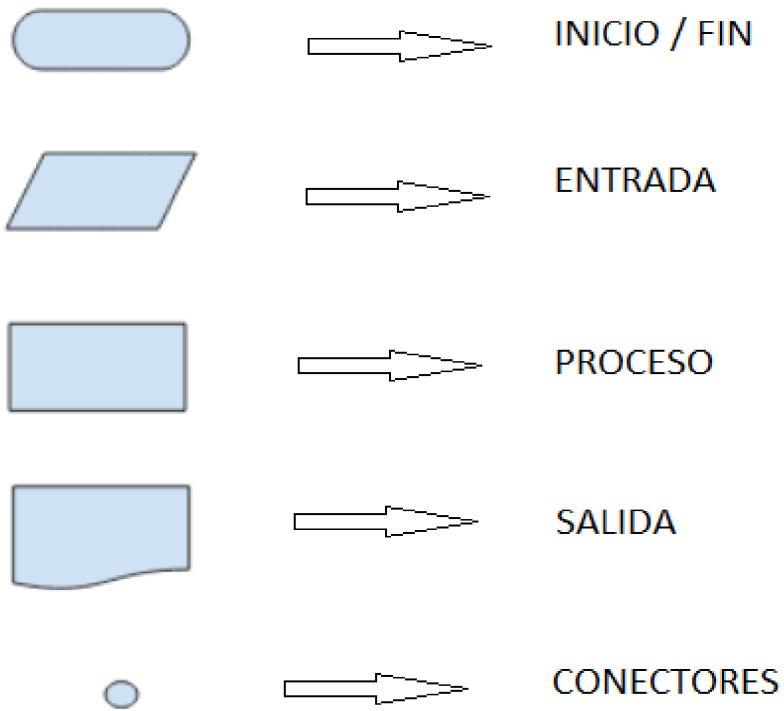


## Representación de algoritmos

Los algoritmos se pueden representar en pseudo (lenguaje formal del humano)

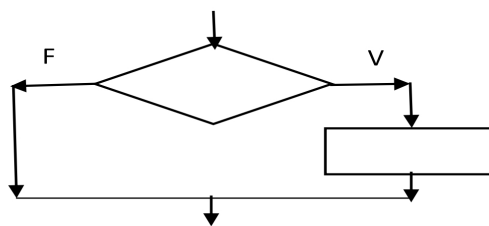
Los algoritmos se pueden representar en Diagramas de Flujo

Diagramas de Flujo:

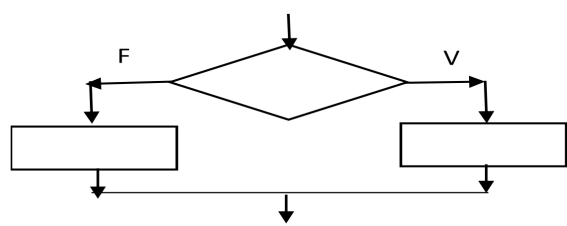


### Condicionales

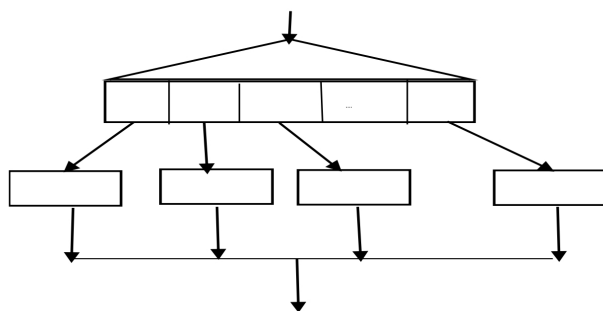
#### CONDICIONAL SIMPLE



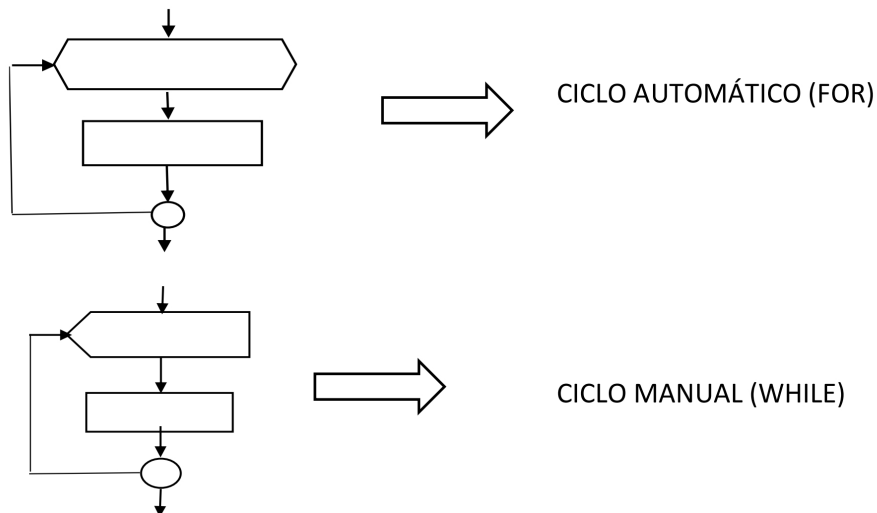
#### CONDICIONAL DOBLE



#### CONDICIONAL MÚLTIPLE



## Repetitivos



Ejercicio 1. Dado un número de 5 dígitos, eliminar el dígito central

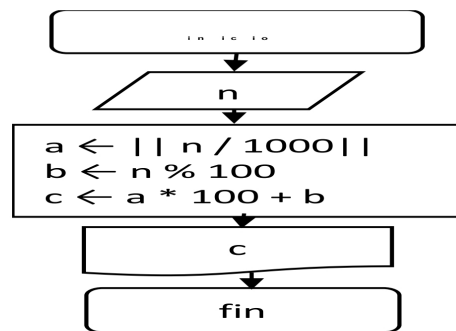
El algoritmo en pseudo, queda:

INICIO

```
read (n);
a ← || n / 1000 ||
b ← n % 100
c ← a * 100 + b
print (c);
```

FIN

El algoritmo en Diagrama de Flujo, queda:



Ejercicio 2. Dado un número de 6 dígitos, eliminar el dígito que representa el dígito de la decena

Ejemplo: Establecemos un numero cualquiera de 6 dígitos, en lo posible un buen ejemplo con dígitos diferentes para poder analizar y lograr descubrir las expresiones aritméticas

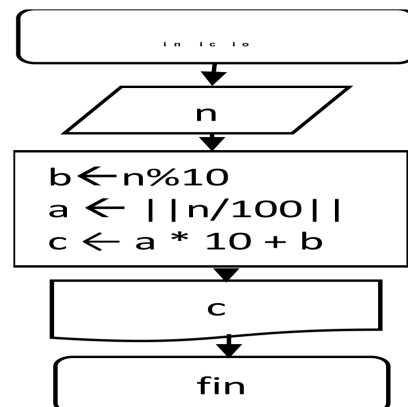
El algoritmo en pseudo, queda:

INICIO

```
read (n);
b ← n % 10
a ← || n / 100 ||
c ← a * 10 + b
print (c);
```

FIN

El algoritmo en Diagrama de Flujo, queda:



3. Resolver la siguiente expresión  $a \leftarrow b * c / d$ 

Entrada	proceso	salida
b, c, d	$a \leftarrow b * c / d$ $d < > 0$	a

Pseudo:

Inicio

Read b, c, d

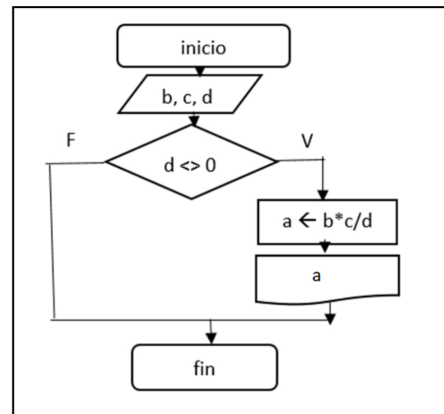
```

If d < > 0 then a = b*c/d
print a

```

Fin

## Diagrama de Flujo



## 4. Encontrar el menor de 3 números

Entrada	proceso	salida
4, 9, 3	¿?	3

Diseño de variables:

a	b	c	men
4	9	3	3

Pseudo:

Inicio

Read a, b, c

```

If a < b then men = a
else men = b

```

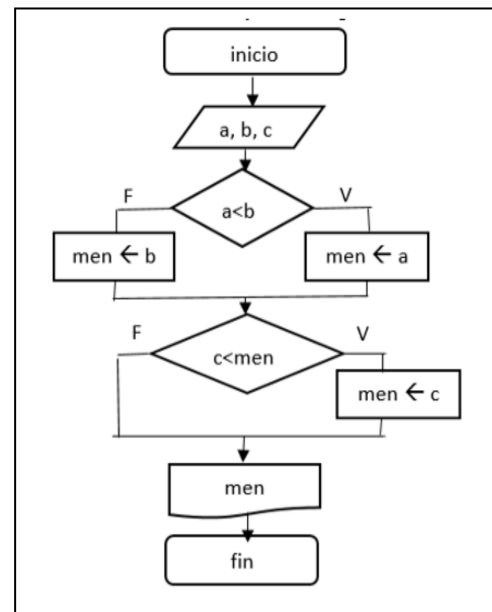
```

if c < men then men = c
print men

```

Fin

## Diagrama de Flujo



Se considera nidos de condiciones:

