



El futuro digital
es de todos

MinTIC

Formulación y Diagramación de soluciones

Estructuras de Control

»
Misión TIC 2022

xxx



Misión
TIC 2022



El futuro digital
es de todos

MinTIC

Temas – Sesión 1



● Formulación y Diagramación de soluciones

- Operadores aritméticos - relacionales
- Operadores Lógicos

● Estructuras de Control

- Condicionales
- Ciclos



Operadores aritméticos

Fundamentos de Programación - Python



El futuro digital
es de todos

MinTIC



x
x
x

Python



Operadores aritméticos - **Jerarquía** – Refuerzo Semana 1



De izquierda a derecha

Operador	Nombre	Nivel Jerárquico
()	Paréntesis	1er nivel
**	Potencia	2do nivel
*	Multiplicación	3er nivel
/ //	División	
+	Suma	4to nivel
-	Resta	

%

Módulo (Residuo)



Práctica Operadores aritméticos (Jerarquía)

```
Python Desactivar audio Detener video Seguridad
Type "help", "copyright",
>>> 7/2
3.5
>>> 7//2
3
>>> 7%2
1
>>> 7+(5*2)
17
>>> 7+5*2
17
>>> 5+10/2-4*3
-2.0
>>> n1=40
>>> n1
40
>>> n2=15
>>> type(n1)
<class 'int'>
>>> type(n2)
<class 'int'>
>>> n3=n1/n2
>>> n3
2.6666666666666665
>>> type(n3)
<class 'float'>
>>>
```



Operadores Relacionales

SÍMBOLO	DESCRIPCIÓN	EJEMPLO	BOOLEANO
==	IGUAL QUE	5 == 7	FALSE
!=	DISTINTO QUE	ROJO != VERDE	TRUE
<	MENOR QUE	8 < 12	TRUE
>	MAYOR QUE	12 > 7	TRUE
<=	MENOR O IGUAL QUE	16 <= 17	TRUE
>=	MAYOR O IGUAL QUE	67 >= 72	FALSE



Práctica Operadores Relacionales

```
Python 3.9.2 (tags/v3.9.2
Type "help", "copyright",
>>> 7>5
True
>>> 5==7
False
>>> 5!=3
True
>>> 56<45
False
>>>
```



x
x
x



Operadores Lógicos

Operador AND



Tabla de verdad de la conjunción

P	q	$P \wedge q$
V	V	V
V	F	F
F	V	F
F	F	F

Operador OR



Tabla de verdad de disyunción o suma logica

P	q	$P \vee q$
V	V	V
V	F	V
F	V	V
F	F	F

Operador NOT



Práctica Operadores lógicos

x
x
x

```
Python 3.9.2 (tags/v3.9.2:13f6953, Oct 7 2021)
Type "help", "copyright", "credits() or "license()" for more
>>> 2<3
True
>>> 7>5
True
>>> 2<3 and 7>5
True
>>> 4==5
False
>>> 2<3 and 4==5
False
>>> 2<3 or 7>5
True
>>> 2<3 or 6==7
True
>>> 3<2 or 6==7
False
>>>
```

Estructuras de Control : Condicionales



El futuro digital
es de todos

MinTIC



Estructura de Control Condicional - Conceptualización



Conceptualización



Deseas ir hacia el trabajo, normalmente lo haces en bus que se tarda 45 minutos. pero te levantaste un poco tarde y solo faltan 15 minutos para el ingreso a la oficina, tienes entonces dos caminos:

Opción 1: Ir en bus y llegar tarde.

Opción 2: Ir en taxi y llegar a tiempo.

En vista que no puedes faltar a las normas de tu empresa decides tomar la opción de viajar en taxi y no usar el bus, ¿vez como las condiciones entran en todas los aspectos de nuestra vida cotidiana?, ahora veamos como se vería nuestro condicional en **Pseudo-Código:**

si (tiempo ≥ 45)

Escribir «Tomar el bus»

sino

Escribir «Tomar un Taxi»

Fin-si

Estructuras de Control : Condicionales



El futuro digital
es de todos

MinTIC



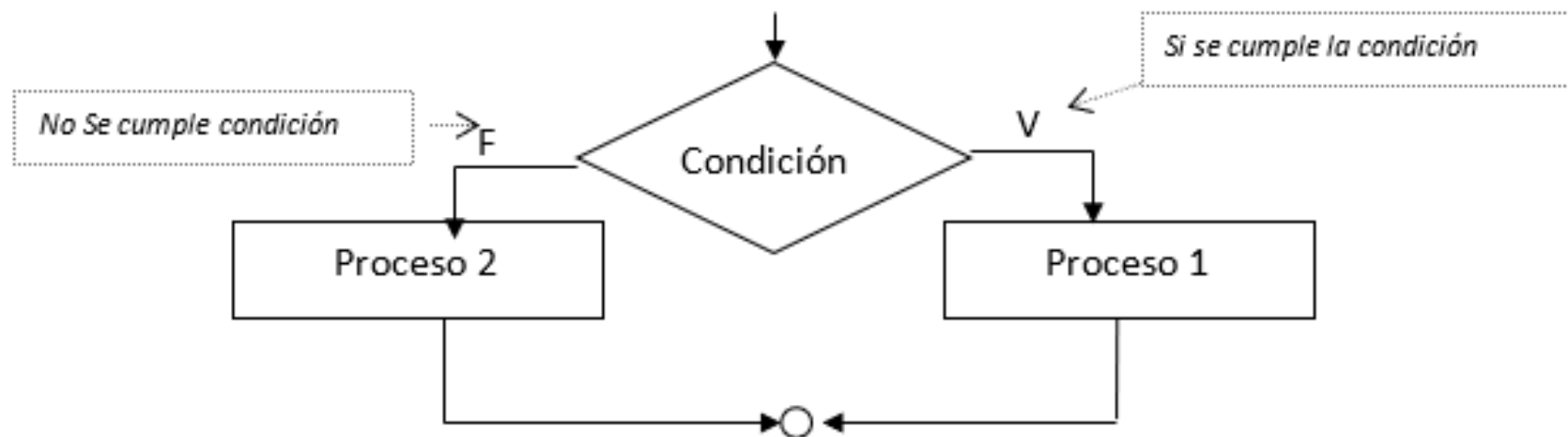
Condicional Simple

Situaciones o problemas de manejo de información con UNA sola condición



Estructuras de Control : Condicionales

Condicional Simple



Esta estructura se utiliza dentro de un proceso, cuando se presenta una decisión, condición o pregunta, si se cumple la condición, es decir si es verdadera, se realiza el proceso 1, en caso de no cumplirse la condición, es decir si es falsa se realiza el proceso 2.



Condicional Simple



En pseudocódigo, la estructura del condicional sería:

SI Condición **ENTONCES**

Proceso 1

SINO

Proceso 2

FIN SI



El futuro digital
es de todos

MinTIC



Ejercicios



Conditional Simple

x
x
x



Dado el **nombre** y **salario de un empleado**, calcular el subsidio de transporte, teniendo en cuenta que **si el salario es menor o igual a \$1.000.000 entonces tiene derecho a un subsidio de transporte por valor de \$120.000**, de lo contrario no tiene derecho al subsidio de transporte. Se debe visualizar el **nombre, salario y subsidio de transporte**



El futuro digital
es de todos

MinTIC

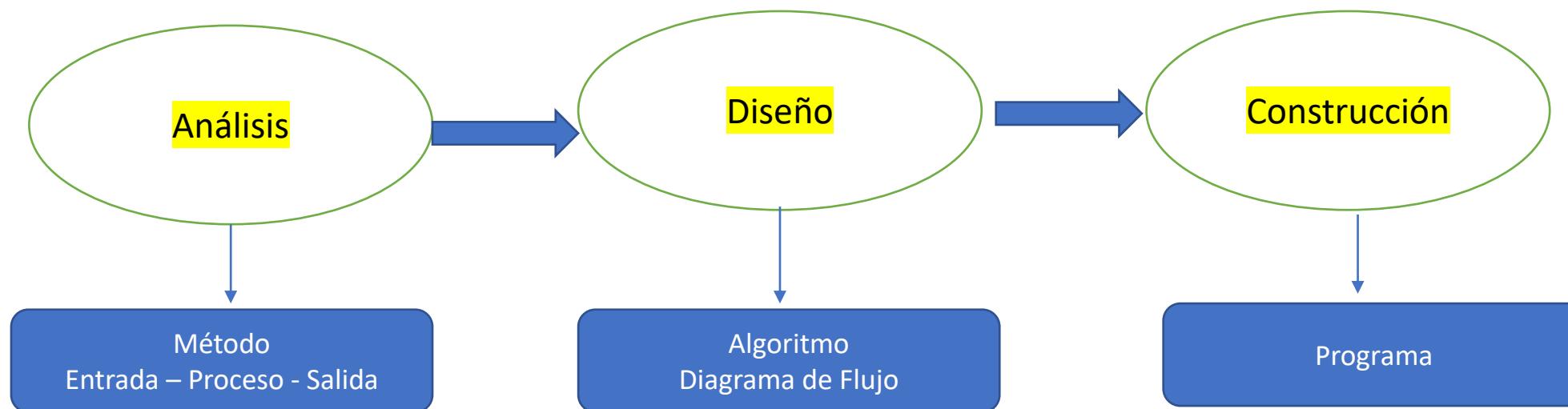


Ejercicios



Condicional Simple

Metodología -> Pensamiento lógico estructurado





El futuro digital
es de todos

MinTIC



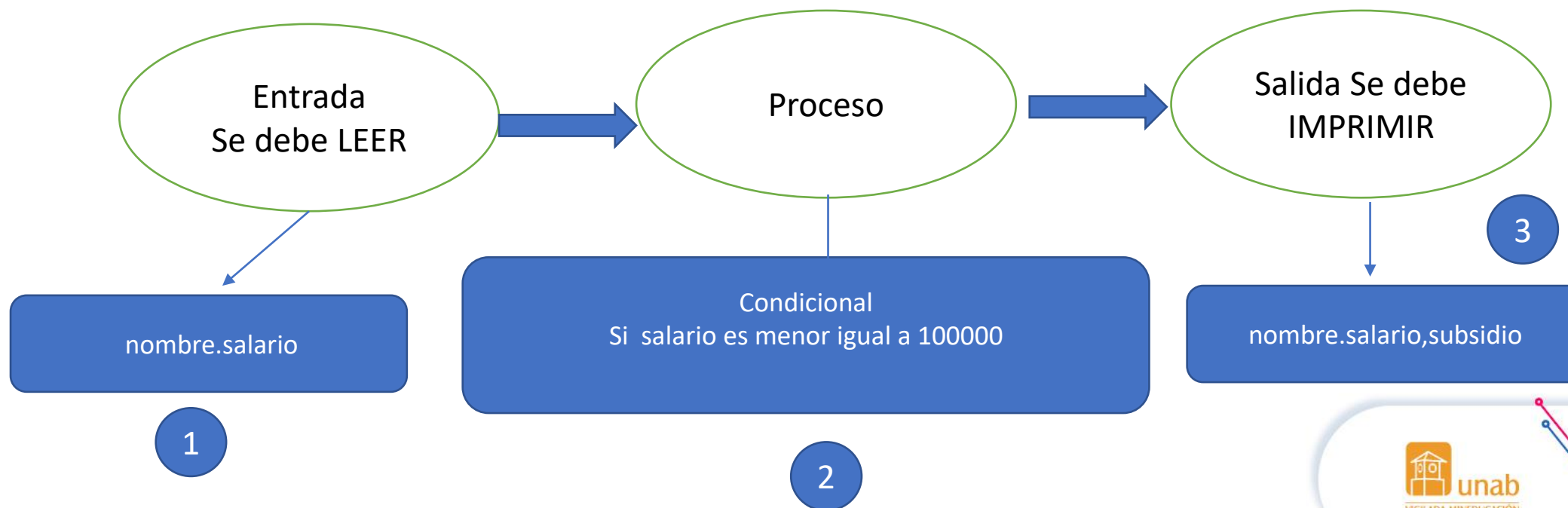
Ejercicios



Condicional Simple

Análisis → Método Entrada-Proceso-Salida

Operaciones, cálculos, **estructuras de control**





El futuro digital
es de todos

MinTIC



Ejercicios



Condicional Simple

Diseño → Algoritmo

```
1  Algoritmo subsidio_transporte
2      Escribir "Nombre empleado: "
3      Leer nombre
4      Escribir "Salario: "
5      Leer salario
6      Si salario ≤ 1000000 Entonces
7          subsidio = 120000
8      SiNo
9          subsidio = 0
10     Fin Si
11     Escribir "Nombre empleado: ", nombre
12     Escribir "Salario: ", salario
13     Escribir "Subsidio de transporte: ",
14 FinAlgoritmo
```

PSelnt - Ejecutando proceso SUBI...

*** Ejecución Iniciada. ***

Nombre empleado:

> Sergio Medina

Salario:

> 950000

Nombre empleado: Sergio Medina

Salario: 950000

Subsidio de transporte: 120000

*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible



El futuro digital
es de todos

MinTIC

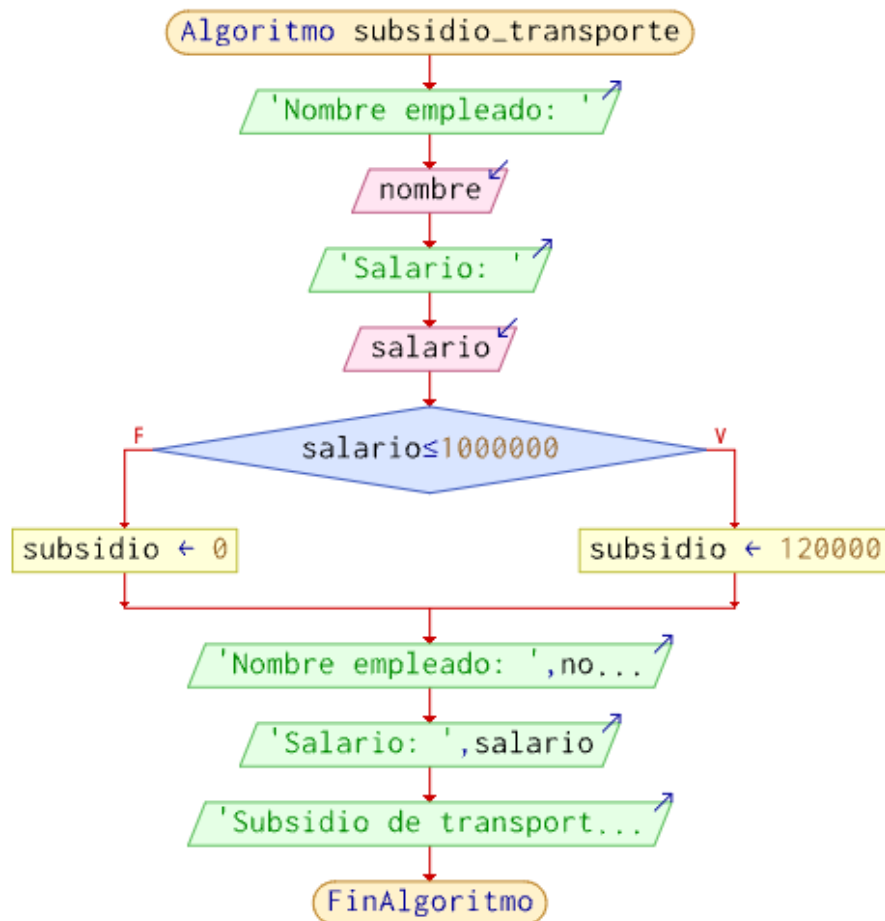


Ejercicios



Condicional Simple

Diseño → Diagrama de flujo





El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

Condicional Simple

Construcción → Programa

```
# Programa para calcular subsidio de transporte
# Autor: Sergio Medina
# Fecha: 25/04/2022

nombre=input("Nombre empleado: ")
salario=float(input("Salario: "))

if salario<=1000000:
    subsidio=120000
else:
    subsidio=0

print("Nombre empleado: ",nombre)
print("Salario: ", '{:,.2f}'.format(salario))
print("Subdisio de transporte: ", '{:,.2f}'.format(subsidio))
```

Ejercicios



Estructuras de Control : Condicionales



El futuro digital
es de todos

MinTIC



Condicional Anidado

Situaciones o problemas de manejo de información con varias condiciones, unas dentro de otras.

Estructuras de Control : Condicionales



El futuro digital
es de todos

MinTIC

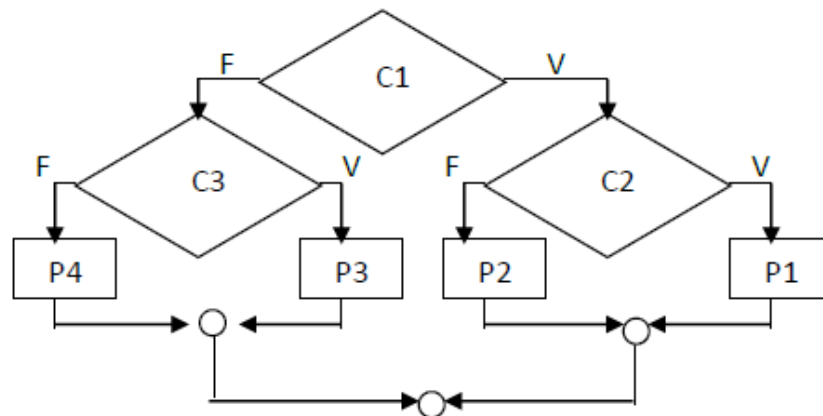


Condicional Anidado



También podemos encontrar CONDICIONALES ANIDADADOS, es decir un condicional dentro de otro.

Gráficamente sería



Donde,

C1, C2 y C3 son las condiciones ó preguntas.

P1, P2, P3 y P4 son los procesos.

En caso de que la C1 sea verdadera se evalúa la C2 y si esta es verdadera se realiza el proceso 1, de lo contrario se realiza el proceso 2. Si la C1 es falsa entonces se evalúa la C3 y así sucesivamente.



El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

Ejercicios



Condicional Anidado

Dado el **nombre del estudiante** y la **calificación cuantitativa** de una evaluación (0-100), se pide hallar la calificación cualitativa, de acuerdo a la siguiente información

Rango Evaluación	Evaluación cualitativa
0 - 59	D
60 - 79	C
80 - 89	B
90 - 100	A

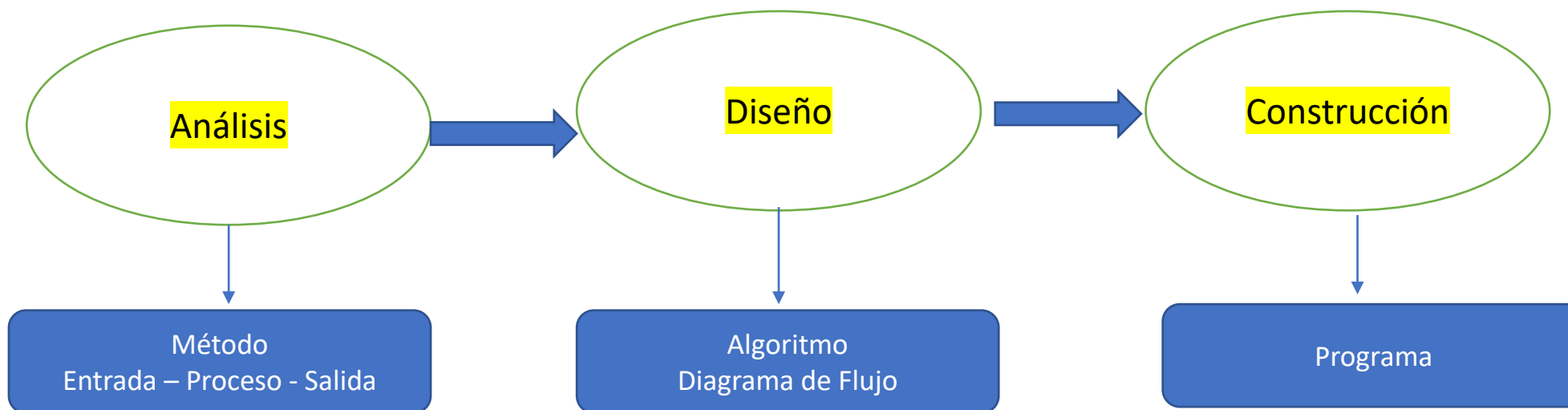
Se pide visualizar, **nombre**, **calificación cuantitativa** y **cualitativa**.



Condicional Anidado



Metodología -> Pensamiento lógico estructurado





El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

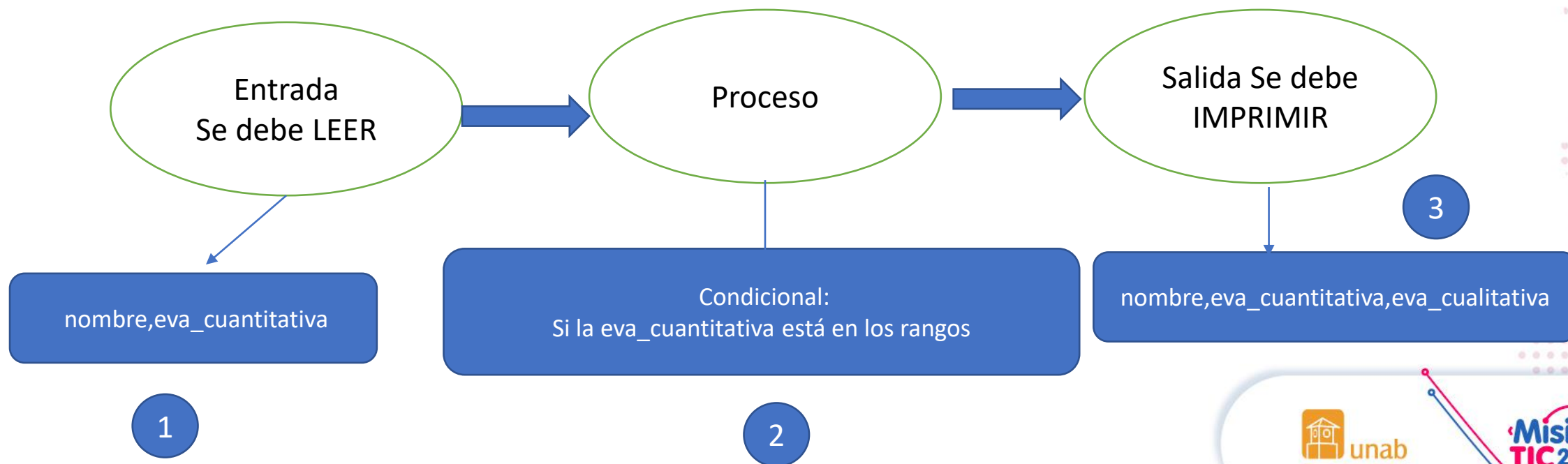
Ejercicios



Condicional Anidado

Análisis → Método Entrada-Proceso-Salida

Operaciones, cálculos, **estructuras de control**





El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

Condicional Anidado

Ejercicios



```
1 Algoritmo calificacion_cualitativa
2   Escribir "Nombre estudiante: "
3   Leer nombre
4   Escribir "Nota Cuantitativa (0-100): "
5   Leer eva_cuantitativa
6   Si eva_cuantitativa ≤ 59 Entonces
7       eva_cualitativa = "D"
8   SiNo
9       Si eva_cuantitativa ≤ 79 Entonces
10          eva_cualitativa = "C"
11       SiNo
12          Si eva_cuantitativa ≤ 89 Entonces
13              eva_cualitativa = "B"
14          SiNo
15              eva_cualitativa ← "A"
16          Fin Si
17       Fin Si
18   Fin Si
19   Escribir "Nombre: ", nombre
20   Escribir "Evaluación Cuantitativa: ", eva_cuantitativa
21   Escribir "Evaluación Cualitativa: ", eva_cualitativa
22 FinAlgoritmo
```

```
PSeInt - Ejecutando proc...
*** Ejecución Iniciada. ***
Nombre estudiante:
> Sergio Medina
Nota Cuantitativa (0-100):
> 75
Nombre: Sergio Medina
Evaluación Cuantitativa: 75
Evaluación Cualitativa: C
*** Ejecución Finalizada. ***
```

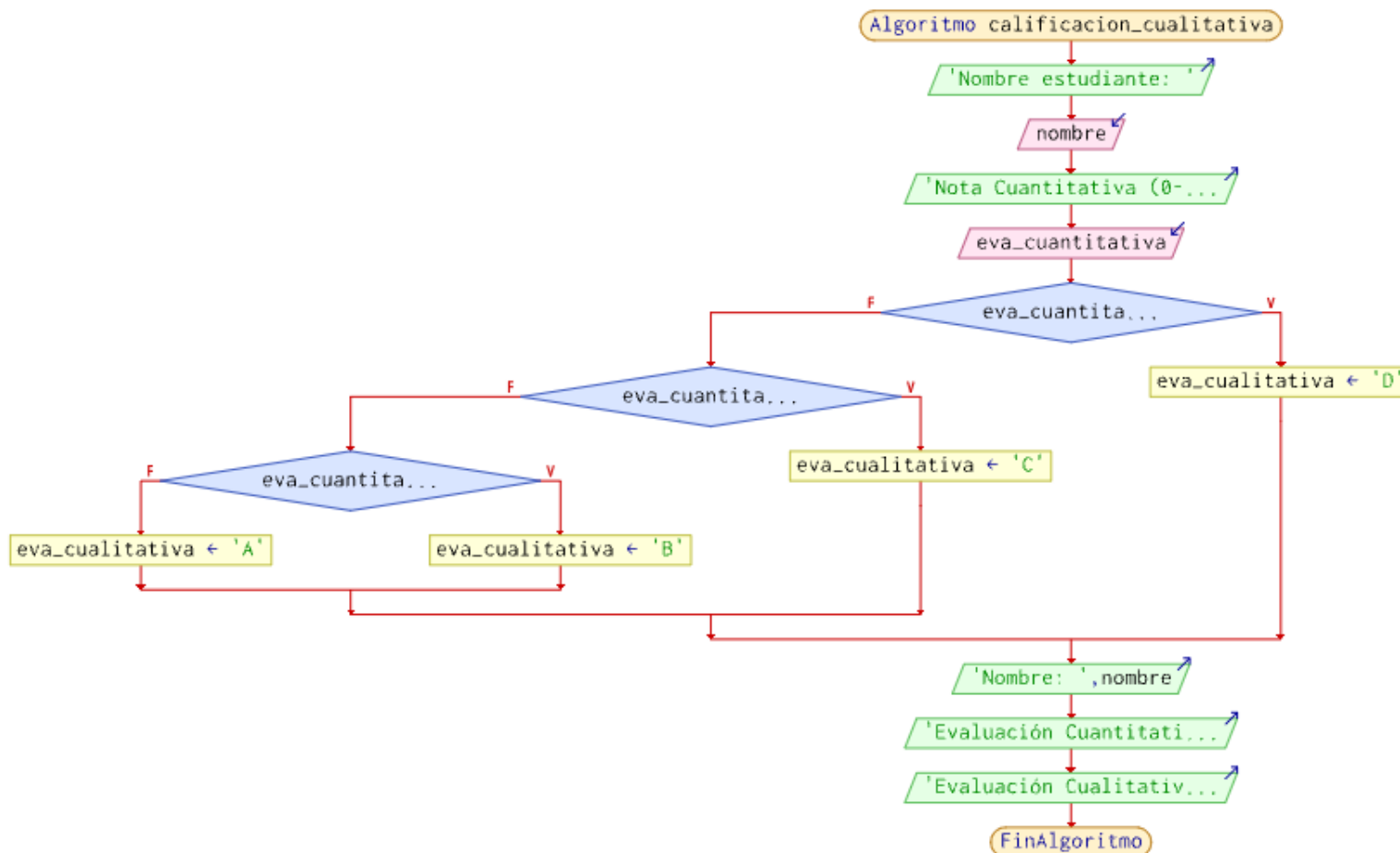
☐ No cerrar esta ventana ☐ Siempre visible

Diseño → Algoritmo



Condicional Anidado

Diseño → Diagrama de flujo





El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

Condicional Anidado

Ejercicios



x

```
# Programa para calcular evaluación cualitativa
# Autor: Sergio Medina
# Fecha: 26/04/2022

nombre=input("Nombre estudiante: ")
eva_cuantitativa=int(input("Evaluación Cuantitativa: "))

if eva_cuantitativa<=59:
    eva_cualitativa="D"
elif eva_cuantitativa<=79:
    eva_cualitativa="C"
elif eva_cuantitativa<=89:
    eva_cualitativa="B"
else:
    eva_cualitativa="A"

print("Nombre estudiante: ",nombre)
print("Evaluación Cuantitativa: ",eva_cuantitativa)
print("Evaluación Cualitativa: ",eva_cualitativa)
```

Construcción → Programa



Condicional Anidado



Dado el **nombre** y **estrato (1,2,3,4,5)** de un usuario del servicio de energía eléctrica, calcular lo que pagaría de tarifa básica del servicio de energía eléctrica, que depende del estrato, así

Estrato	Tarifa Básica
1	\$10.000
2	\$15.000
3	\$30.000
4	\$50.000
5	\$65.000

Se pide visualizar el **nombre** y **tarifa básica**

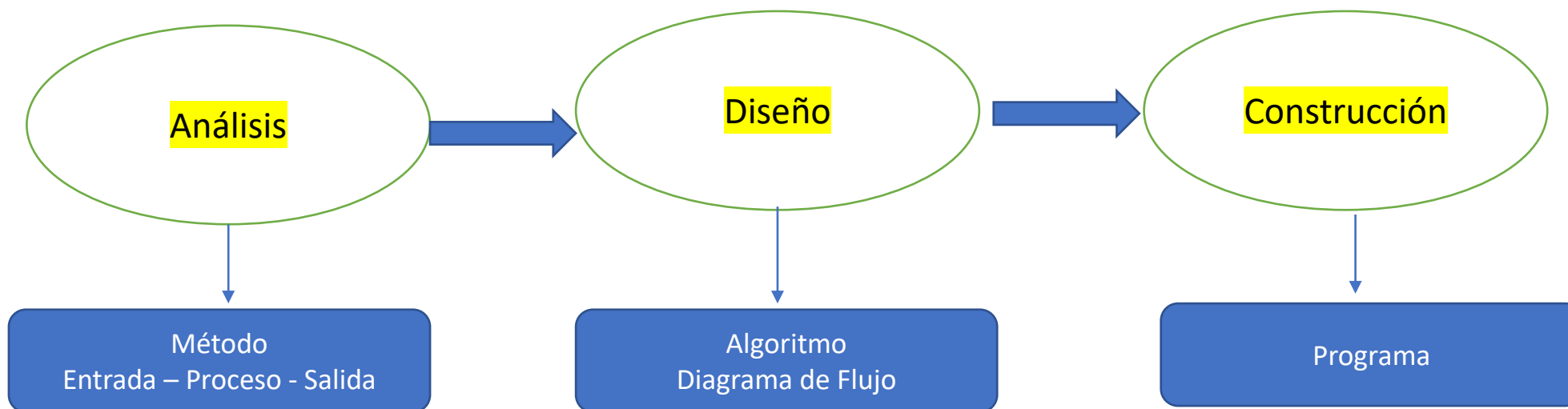




Condicional Anidado



Metodología -> Pensamiento lógico estructurado



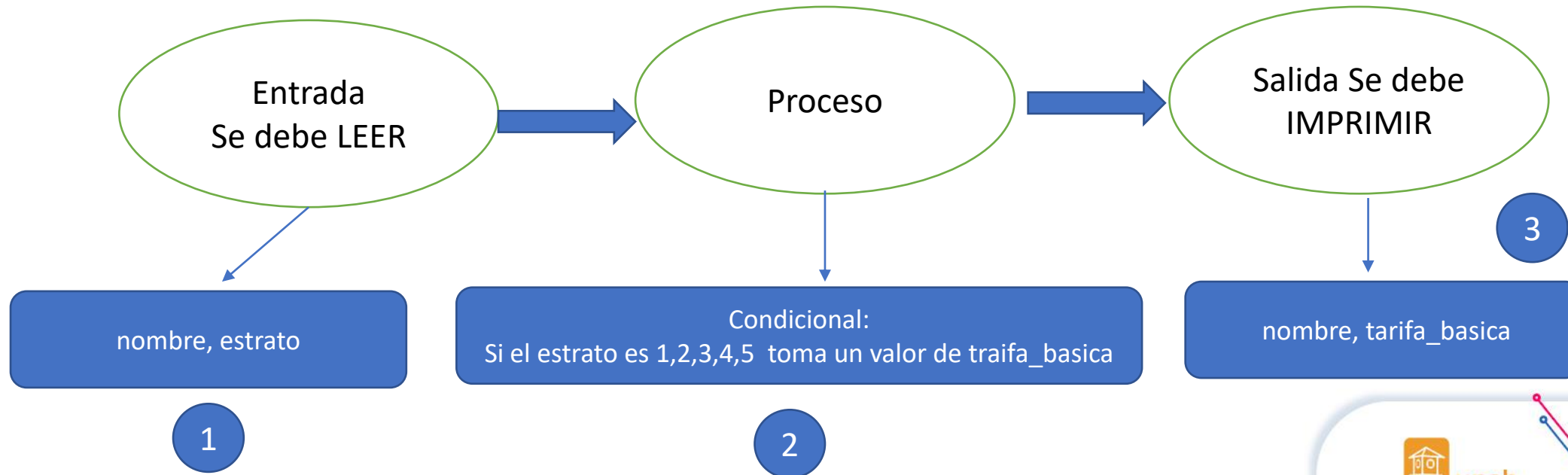


Condicional Anidado



Análisis → Método Entrada-Proceso-Salida

Operaciones, cálculos, **estructuras de control**





El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

Condicional Anidado

Ejercicios



X
X
X

```
Algoritmo Calcular_tarifa_basica
  Escribir "Nombre usuario: "
  Leer nombre
  Escribir "Estrato(1,2,3,4 o 5): "
  Leer estrato
  Si estrato=1 Entonces
    tarifa_basica=10000
  SiNo
    Si estrato=2 Entonces
      tarifa_basica=15000
    SiNo
      Si estrato=3 Entonces
        tarifa_basica=30000
      SiNo
        Si estrato=4 Entonces
          tarifa_basica=50000
        SiNo
          tarifa_basica=65000
        Fin Si
      Fin Si
    Fin Si
  Fin Si
  Escribir "Nombre usuario: ", nombre
  Escribir "Tarifa Básica: ", tarifa_basica
FinAlgoritmo
```

```
PSeInt - Ejecutando pr...
*** Ejecución Iniciada. ***
Nombre usuario:
> Sergio Medina
Estrato(1,2,3,4 o 5):
> 4
Nombre usuario: Sergio Medina
Tarifa Básica: 50000
*** Ejecución Finalizada. ***
☐ No cerrar esta ventana ☐ Siempre visible 
```

Diseño → Algoritmo



El futuro digital
es de todos

MinTIC

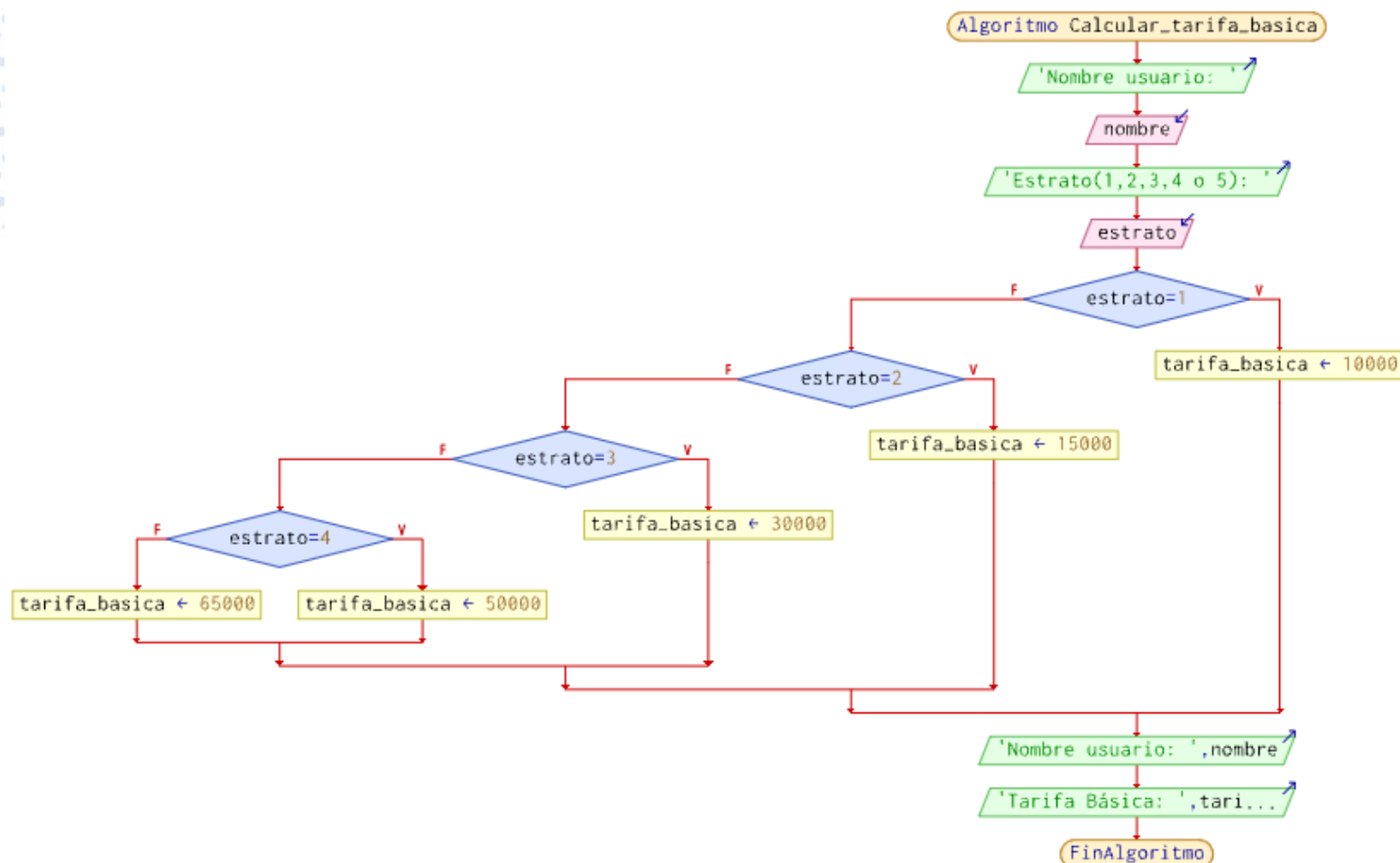


Estructuras de Control : Condicionales

Ejercicios



Condicional Anidado



Diseño → Diagrama de flujo



El futuro digital
es de todos

MinTIC



Estructuras de Control : Condicionales

Condicional Anidado

Ejercicios



```
# Programa para calcular tarifa básica de un usuario servicio de energía
# Autor: Sergio Medina
# Fecha: 27/04/2022

nombre=input("Nombre del usuario: ")
estrato=int(input("Estrato (1,2,3,4,o 5): " ))

if estrato==1:
    tarifa_basica=10000
elif estrato==2:
    tarifa_basica=15000
elif estrato==3:
    tarifa_basica=30000
elif estrato==4:
    tarifa_basica=50000
else:
    tarifa_basica=65000

print("Nombre usuario: ",nombre)
print("Tarifa Básica: ",'{:,.2f}'.format(tarifa_basica))
```

Construcción → Programa

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Estructuras de Control Iterativas (Ciclos) - Conceptualización



Conceptualización



En la vida diaria se manejan con mucha frecuencia las estructuras iterativas, en donde se encuentra **procesos que se repiten una cantidad de veces**, por ejemplo:

Ejemplo 1: En un entrenamiento con el equipo de baloncesto, el profesor – entrenador nos invita a realizar 5 vueltas a la cancha como calentamiento. En este ejemplo, el proceso que se repite es la vuelta y se debe realizar una cantidad de veces determinada (5). En términos de programación, se le denomina la estructura iterativa contralada por cantidad – FOR.

Ejemplo 2: En una clase de educación física, el profesor es llamado desde la coordinación y nos invita a realizar vueltas a la cancha mientras regresa . En este ejemplo, el proceso que se repite es la vuelta y no se conoce la cantidad a realizar, depende de la condición del regreso del docente, es decir mientras el docente no regrese, debo dar vuelta. En términos de programación, se le denomina la estructura iterativa contralada por condición – WHILE.

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por cantidad - FOR

Situaciones o problemas de manejo de información que incluyen iteraciones o ciclos que se repiten una cantidad de veces determinada o conocida

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC

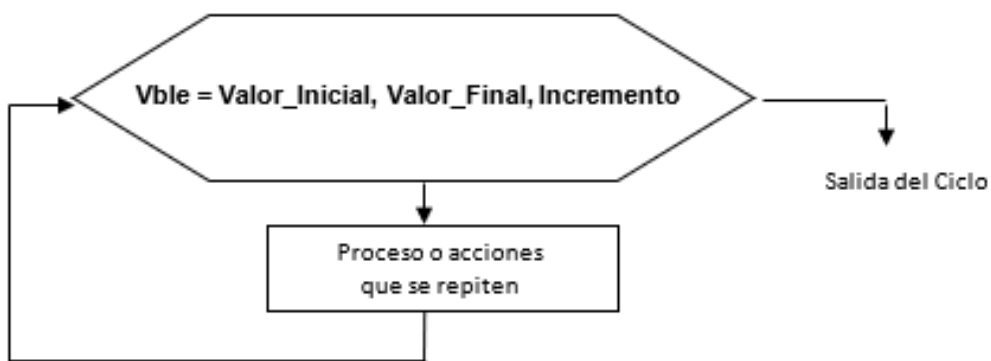


Iteración o ciclos controlados por cantidad - FOR

×

×

×



Vble = Encargada de tomar valores de acuerdo con cada ocurrencia de la repetición. Puede darse cualquier nombre a la variable (tener en cuenta las reglas para identificadores), aunque se utiliza con mucha frecuencia, los nombres de I, J, K entre otros. Las iteraciones estarán controladas por esta variable.

Valor_Inicial = Constante o variable con el valor inicial del ciclo, esta asignación solo se realiza una vez, cuando se inicializa el ciclo, es la primera ocurrencia.

Valor_Final = Constante o variable con el valor final del ciclo, esta determina la condición para terminar el ciclo, es la última ocurrencia.

Incremento = Constante o variable que representa el intervalo de aumento de la variable del ciclo, cuando el incremento es 1 se puede omitir, si es un decremento se representa con un valor negativo.

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por cantidad - FOR



En pseudocódigo, la estructura sería:

PARA Vble= Valor_Inicial **HASTA** Valor_Final **INCREMENTO** INCR

Proceso que se Repite

FIN PARA



Iteración o ciclos controlados por cantidad - FOR



x
x
x

Se tiene la siguiente información sobre los **N (N es suministrado)** usuarios del servicio de AGUA:

- **Código**
- **Nombre**
- **Estado: Puede ser V=Vigente o S=Suspendido**
- **Estrato: Puede ser 1,2,3,4 5 o 6**
- **Consumo del mes (en cm3)**

Se pide **calcular el valor a pagar por concepto de servicio de AGUA de cada usuario, teniendo en cuenta que este valor es la suma del valor de tarifa más el valor del consumo.** También nos indican que el valor de la tarifa básica depende del estrato así:

(ESTRATO - TARIFA BÁSICA): (1 - \$10.000, 2 - \$20.000, 3 - \$30.000, 4 - \$45.000, 5 - \$60.000, 6 - \$70.000)

Además el **valor del consumo es el consumo del mes por el valor de 1 cm3 que de \$200**

Se debe imprimir el **nombre del usuario, el valor de la tarifa básica, el valor del consumo y el valor a pagar por concepto del servicio de AGUA**

NOTA: Se liquida servicio de AGUA a los usuarios con estado V (Vigente)

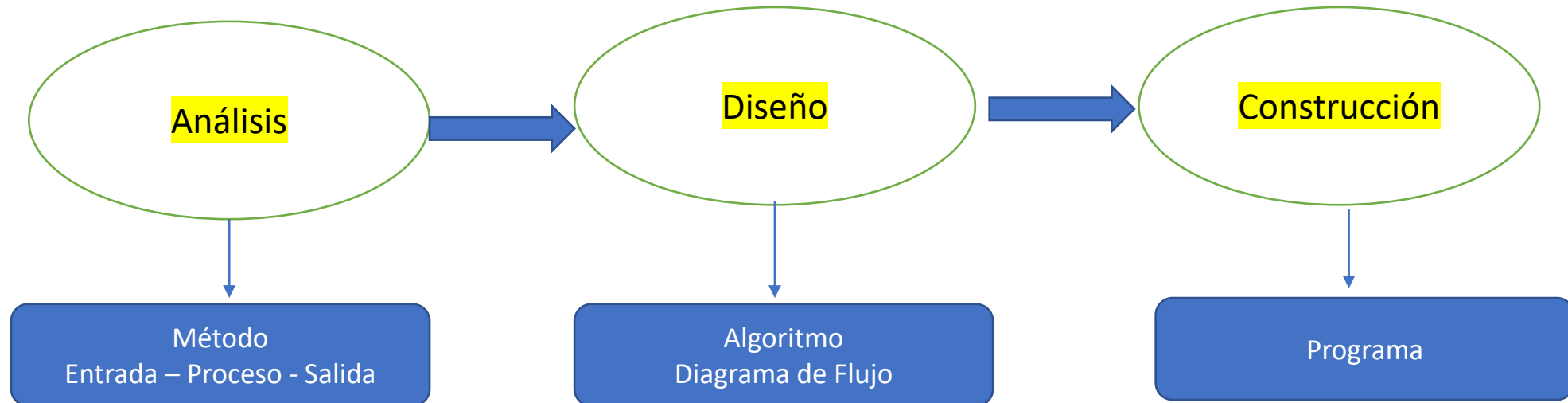




Iteración o ciclos controlados por cantidad - FOR



Metodología -> Pensamiento lógico estructurado





Variables de Control – Estructuras Iterativas

Ejercicios



Iteración o ciclos controlados por cantidad - FOR

Análisis → Método Entrada-Proceso-Salida

Operaciones, cálculos, **estructuras de control**

Entrada
Se debe LEER

Proceso

Salida Se debe
IMPRIMIR

1

N=Cantidad de usuarios

código nombre, estado, estrato,
consumo

3

2

Ciclo: Liquidación de cada usuario (FOR, se conoce la
cantidad de usuarios N)

Condicional: Estado . Si es vigente V se liquida servicio

4

Condicional: Estrato, calcular tarifa básica

5

Calcula valor_consumo y valor_pagar

6

7

nombre, tarifa_básica,
valor_consumo, valor_pagar



Iteración o ciclos controlados por cantidad - FOR



Diseño → Algoritmo

Algoritmo servicio_agua

Escribir "Cantidad de usuarios. "

Leer N

Para i=1 Hasta N Con Paso 1 Hacer

Escribir "Código usuario: "

Leer codigo

Escribir "Nombre: "

Leer nombre

Escribir "Estado (V=Vigente,S=Suspendido): "

Leer estado

Escribir "Estrato: (1,2,3,4,5,6): "

Leer estrato

Escribir "Consumo: "

Leer consumo

```
Si estado="V" Entonces
  Si estrato=1 Entonces
    tarifa_basica=10000
  SiNo
    Si estrato=2 Entonces
      tarifa_basica=20000
    SiNo
      Si estrato=3 Entonces
        tarifa_basica=30000
      SiNo
        Si estrato=4 Entonces
          tarifa_basica=45000
        SiNo
          Si estrato=5 Entonces
            tarifa_basica=60000
          SiNo
            tarifa_basica=70000
          Fin Si
        Fin Si
      Fin Si
    Fin Si
  Fin Si
Fin Si
```

```
valor_consumo=consumo*200
valor_pagar=tarifa_basica+valor_consumo
Escribir "Nombre usuario: ",nombre
Escribir "Tarifa Básica: ",tarifa_basica
Escribir "Valor consumo: ",valor_consumo
Escribir "Valor a pagar: ",valor_pagar
Fin Si
Fin Para
FinAlgoritmo
```



El futuro digital
es de todos

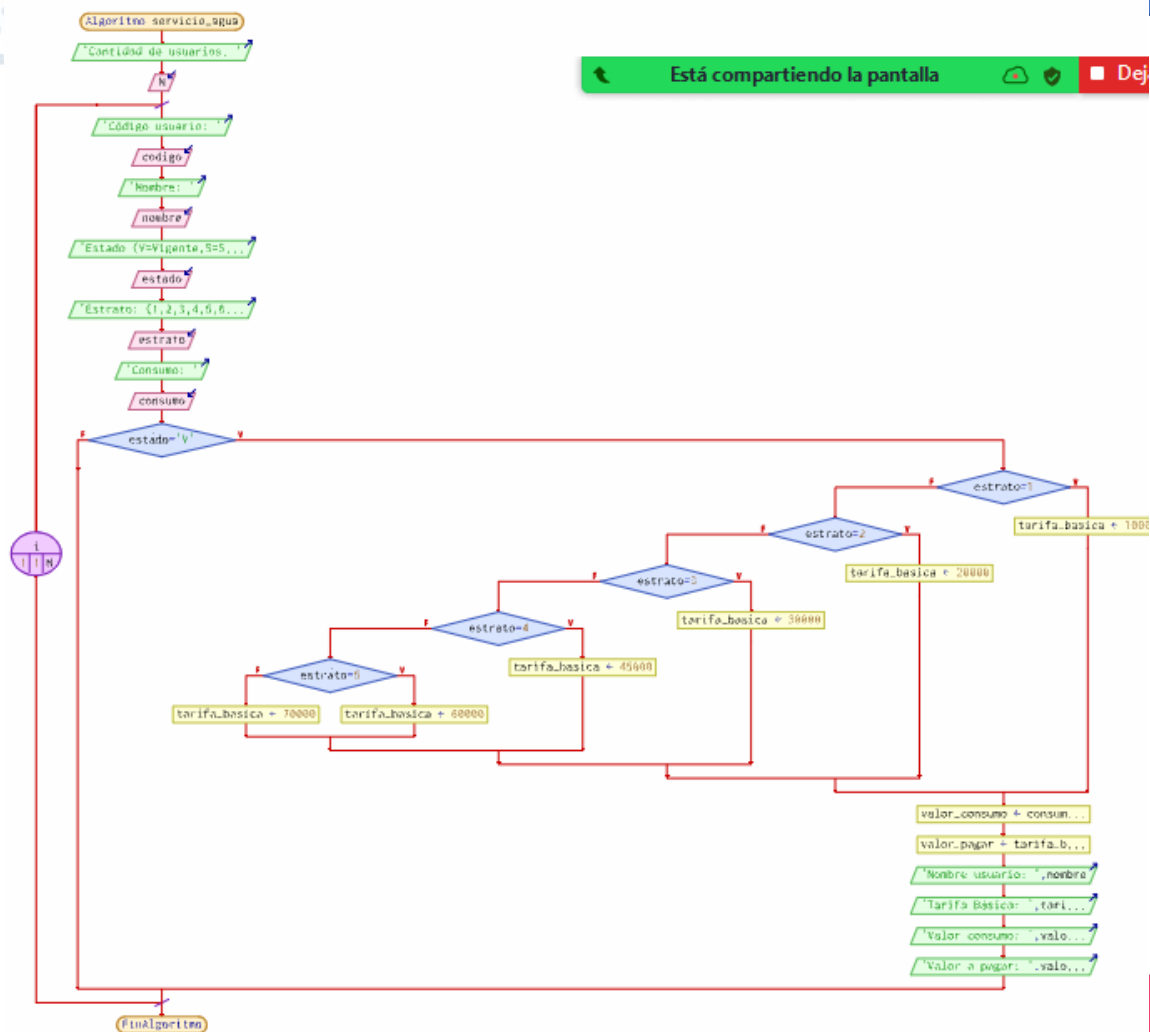
MinTIC



Iteración o ciclos controlados por cantidad - FOR



Diseño → Diagrama de flujo





El futuro digital
es de todos

MinTIC

Variables de Control – Estructuras Iterativas



Iteración o ciclos controlados por cantidad: FOR - RANGE

Programación en Python

```
for i in range(3):  
    print(i)
```

```
# 0  
# 1  
# 2
```

1

```
for i in range(5, 8):  
    print(i, i ** 2)  
    print('end of loop')
```

```
# 5 25  
# 6 36  
# 7 49  
# fin del ciclo
```

2

```
for i in range(10, 0, -2):  
    print(i)
```

```
# 10  
# 8  
# 6  
# 4  
# 2
```

3

Variables de Control – Estructuras Iterativas

Ejercicios



El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por cantidad - FOR



Construcción → Programa

```
# Programa para calcular el valor de servicio de agua
# Autor: Sergio Medina
# Fecha: 3/05/2022

N=int(input("Cantidad de usuarios: "))
for i in range(N):
    codigo=int(input("Código usuario: "))
    nombre=input("Nombre: ")
    estado=input("Estado (V=Vigente,S=Suspendido): ")
    estrato=int(input("Estrato (1,2,3,4,5,6): "))
    consumo=float(input("Consumo del mes: "))
    if estado=="V":
        if estrato==1:
            tarifa_basica=10000
        elif estrato==2:
            tarifa_basica=20000
        elif estrato==3:
            tarifa_basica=30000
        elif estrato==4:
            tarifa_basica=45000
        elif estrato==5:
            tarifa_basica=60000
        else:
            tarifa_basica=70000
    valor_consumo=consumo*200
    valor_pagar=tarifa_basica+valor_consumo
    print("Nombre usuario: ",nombre)
    print("Tarifa básica: ",'{:,.2f}'.format(tarifa_basica))
    print("Valor consumo: ",'{:,.2f}'.format(valor_consumo))
    print("Valor a Pagar: ",'{:,.2f}'.format(valor_pagar))
```



Misión
TIC2022



ACUMULADORES

Son variables de memoria que se incrementan durante la ejecución de un proceso. Este incremento puede ser de dos tipos:

- A través de un valor fijo o constante, normalmente 1, y se denomina **CONTADOR**. Su incremento sería por ejemplo $CONT = CONT + 1$, en donde la variable **CONT**, incrementa en 1 el valor que tiene almacenado.
- A través de otra variable, con lo cual recibe el nombre de **SUMADOR**. Por ejemplo, el total de netos de nómina es la suma de los netos de cada empleado. Sería así:
 $TOTAL_NETOS = TOTAL_NETOS + NETO_EMPLEADO$, en donde el **TOTAL_NETOS** se incrementa con el valor de la variable **NETO_EMPLEADO**, que es un valor variable.

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Variables de Control: Acumuladores



Estas variables ACUMULADORAS siempre se definen sobre procesos que manejen estructuras iterativas, como el caso del FOR, WHILE. Además, durante la ejecución de un proceso manejan tres (3) actividades importantes, que son:

1. Se les debe asignar un valor inicial, normalmente cero (0), pero ello depende del problema a resolver
2. Se les debe incrementar, con el valor fijo, en caso del CONTADOR, o con un valor variable en el caso del SUMADOR.
3. En alguna parte del proceso se conoce su valor final.

Si esquematizamos las estructuras de repetición, mencionadas anteriormente, encontramos tres zonas básicas, que son: Antes, dentro y después del ciclo.

Estas zonas, combinadas con las actividades que maneja un ACUMULADOR, nos permiten concluir:

1. Antes del ciclo, se debe dar valores iniciales a los acumuladores
2. Dentro del ciclo, se debe incrementar los acumuladores.
3. Después del ciclo, se conoce el valor final de un acumulador.



Variables de Control: Acumuladores



Situación problema: Liquidación de Honorarios Docente

Se tiene la siguiente información de los **N docentes** de una institución educativa:

- **Documento de identidad**
- **Nombre**
- **Categoría docente(A,B o C)**
- **Horas laboradas en el mes**

También suministran el valor de la hora que la institución paga a los docentes, dependiendo de su categoría, así:

(Categoría – Valor hora): (A - \$25.000, B - \$35.000, C - \$50.000)

Con base en la información suministrada se pide:

- **Valor a pagar por honorarios para cada docente**
- **Valor total a pagar (Todos los docentes)**
- **Cantidad de docentes de cada una de las categorías.**

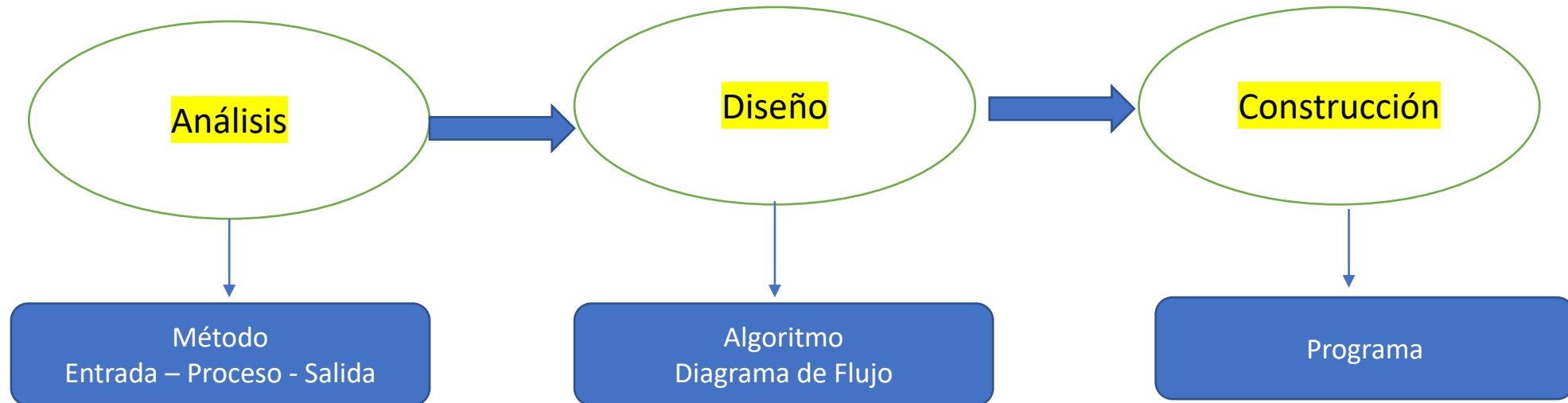




Variables de Control: Acumuladores



Metodología -> Pensamiento lógico estructurado





Variables de Control: Acumuladores



Análisis → Método Entrada-Proceso-Salida

Operaciones, cálculos, **estructuras de control**

Entrada
Se debe LEER

Proceso

Salida Se debe
IMPRIMIR

1

N=Cantidad de docentes,
inicializar acumuladores

2

Ciclo: liquidación honorarios cada docente (FOR,
porque se conoce N docentes)

3

documento, nombre,
categoría, horas

4

Condional: Calcular honorarios con base en la
categoría (Aplicar el valor de horas a las horas
laboradas), incrementar acumuladores

5

nombre, honorarios

6

total_honorarios: Acumlador-Sumador,
variable que suma: honorarios

7

cont_a, cont_b, cont_c, Acumuladores de
tipo contador



Variables de Control: Acumuladores

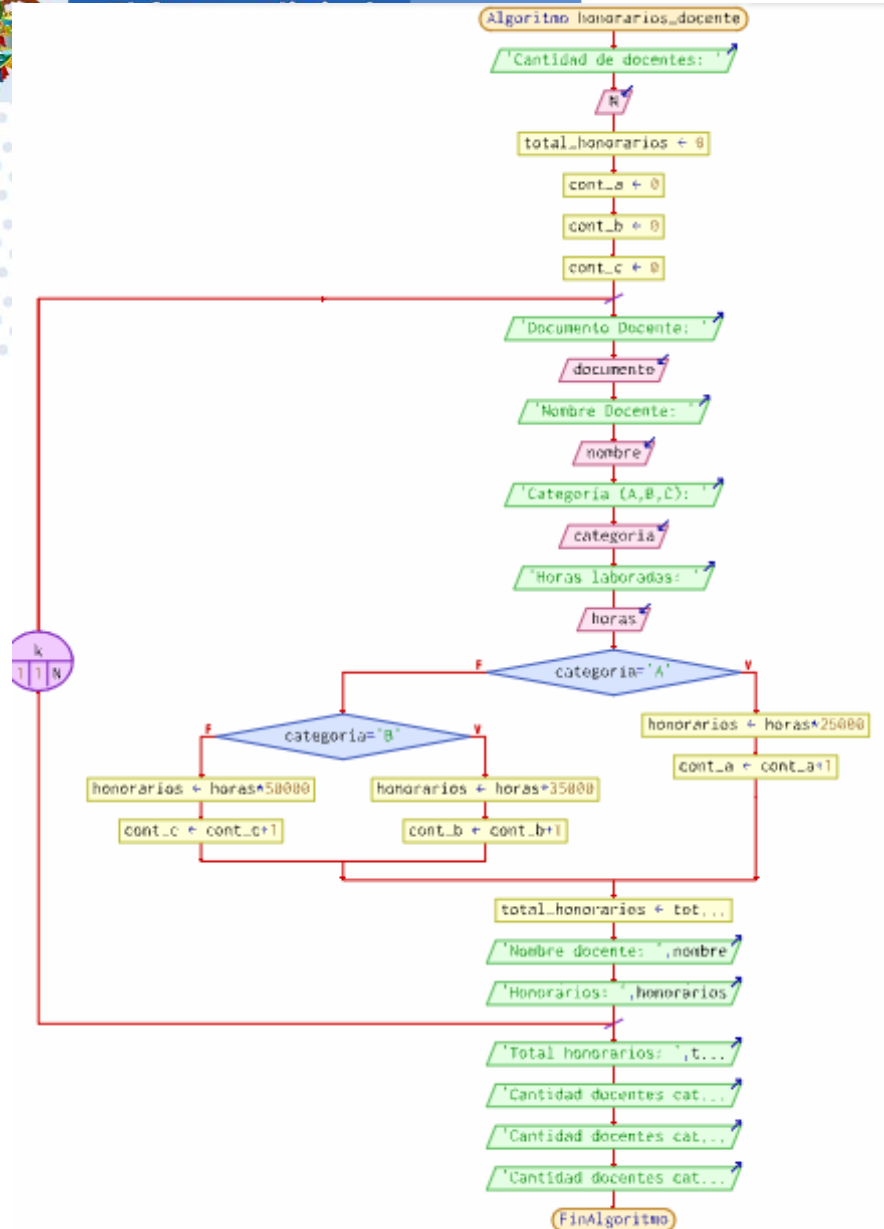
Diseño → Algoritmo



```
Algoritmo honorarios_docente
  Escribir "Cantidad de docentes: "
  Leer N
  total_honorarios=0
  cont_a=0
  cont_b=0
  cont_c=0
  Para k<-1 Hasta N Con Paso 1 Hacer
    Escribir "Documento Docente: "
    Leer documento
    Escribir "Nombre Docente: "
    Leer nombre
    Escribir "Categoria (A,B,C): "
    Leer categoria
    Escribir "Horas laboradas: "
    Leer horas
    Si categoria="A" Entonces
      honorarios=horas*25000
      cont_a=cont_a+1
    SiNo
      Si categoria="B" Entonces
        honorarios=horas*35000
        cont_b=cont_b+1
      SiNo
        honorarios=horas*50000
        cont_c=cont_c+1
      Fin Si
    Fin Si
    total_honorarios=total_honorarios+honorarios
    Escribir "Nombre docente: ",nombre
    Escribir "Honorarios: ",honorarios
  Fin Para
  Escribir "Total honorarios: ",total_honorarios
  Escribir "Cantidad docentes categoria A: ",cont_a
  Escribir "Cantidad docentes categoria B: ",cont_b
  Escribir "Cantidad docentes categoria C: ",cont_c
FinAlgoritmo
```

Variables de Control: Acumuladores

Diseño → Diagrama de flujo



Variables de Control – Estructuras Iterativas

Ejercicios



El futuro digital

MinTIC

```
# Programa para calcular honorarios docente
# Autor: Sergio Medina
# Fecha: 04/05/2022
```

```
N=int(input("Cantidad de docentes: "))
total_honorarios=0
cont_a=0
cont_b=0
cont_c=0
for k in range(N):
    documento=int(input("Documento Docente: "))
    nombre=input("Nombre Docente: ")
    categoria=input("Categoría (A,B,C): ")
    horas=int(input("Horas laboradas: "))
    if categoria=="A":
        honorarios=horas*25000
        cont_a+=1
    elif categoria=="B":
        honorarios=horas*35000
        cont_b+=1
    else:
        honorarios=horas*50000
        cont_c+=1
    total_honorarios+=honorarios
    print("Nombre docente: ",nombre)
    print("Honorarios: ", '{:,.2f}'.format(honorarios))
print("Total honorarios: ", '{:,.2f}'.format(total_honorarios))
print("Cantidad docentes categoría A: ",cont_a)
print("Cantidad docentes categoría B: ",cont_b)
print("Cantidad docentes categoría C: ",cont_c)
```

Está compar

Variables de Control: Acumuladores



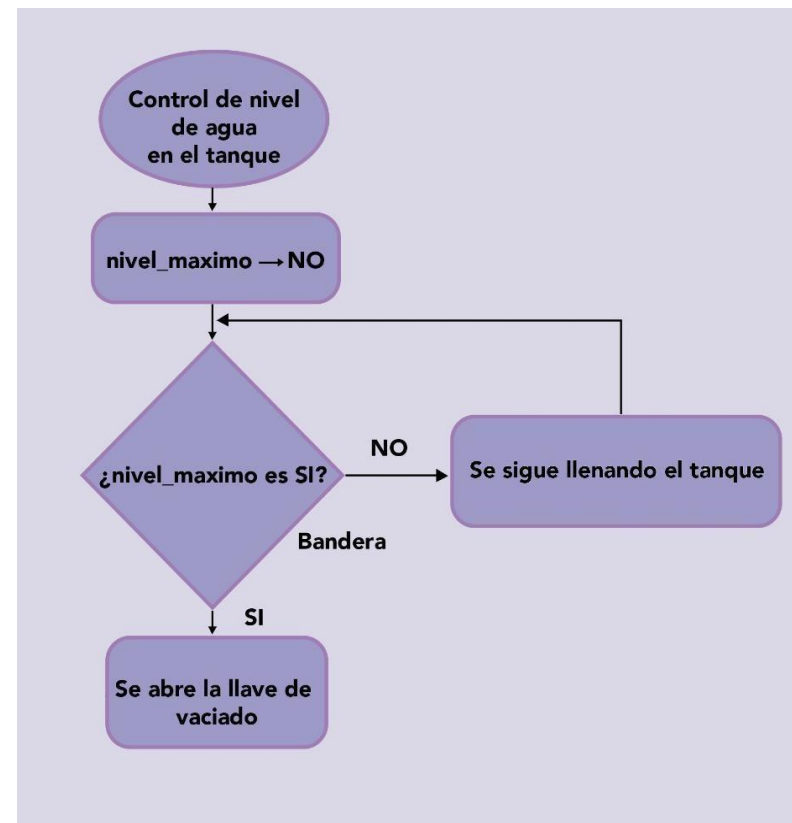
Construcción → Programa



Variables de Control: Banderas

Las **Banderas** hacen referencia a variables que toman un valor, preferiblemente **binario**, booleano e indican un estado; su valor y el cambio del mismo, definen el estado en el que se encuentra el programa.

Por ejemplo, se requiere llevar el **control del nivel de agua de un tanque**.



Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por una condición - WHILE

Situaciones o problemas de manejo de información que incluyen iteraciones o ciclos que se repiten dependiendo de una condición

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

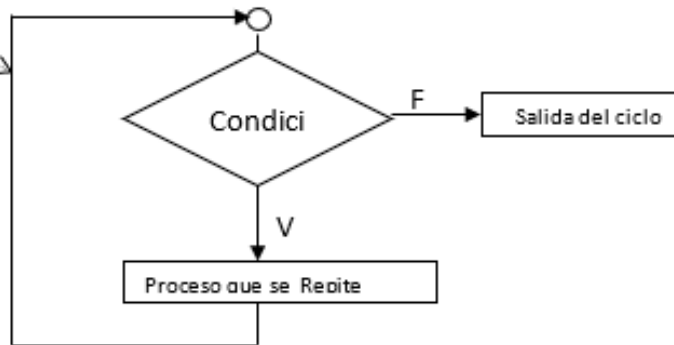
MinTIC



Iteración o ciclos controlados por condicionales - WHILE



El proceso se repite
hasta cuando no se
cumpla la condición



Una característica importante de esta estructura es que está conformada de dos elementos fundamentales:

- Una Condición
- Un Proceso que se repite

Y además se tiene como norma que PRIMERO se realiza la condición y SI SE CUMPLE entonces se realiza como SEGUNDA medida el proceso. Por ello esta estructura se debe utilizar cuando se tengan las siguientes características:

1. Debe existir un proceso que se repita (CICLO).
2. La repetición del proceso debe depender de una condición.

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por condicionales - WHILE

En pseudocódigo:

MIENTRAS Condición **HAGA**

Proceso que se repite

FIN MIENTRAS

Variables de Control – Estructuras Iterativas



El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por condicionales - WHILE



GUÍA del WHILE (Iteración manejada con condición)

En la estructura de control WHILE se presenta una DOBLE lectura de información, por las siguientes razones:

- La primera lectura se encuentra ANTES del ciclo, debido a que, al ingreso al ciclo WHILE, se encuentra una condición que necesita tener un valor en la variable de control bandera.
- La segunda lectura se encuentra DENTRO del ciclo, al final del proceso que se repite, debido a que es necesario, una vez procesada una ocurrencia de la repetición, pasar a la siguiente información almacenada en la variable bandera, es decir, pasar a leer el siguiente elemento. Si no se realiza esta lectura del siguiente elemento se podría incurrir en un ciclo SIN FIN, por no tener variación en la variable de control necesaria para el ingreso o salida del ciclo

Guía del WHILE



Iteración o ciclos controlados por condicional - WHILE



Dada una lista de **números enteros**, se pide calcular e imprimir:

- **Cuáles y cuántos números son pares**
- **Cuáles y cuántos números son impares**

La **lista se termina cuando el número ingresado es -1 (Bandera)**

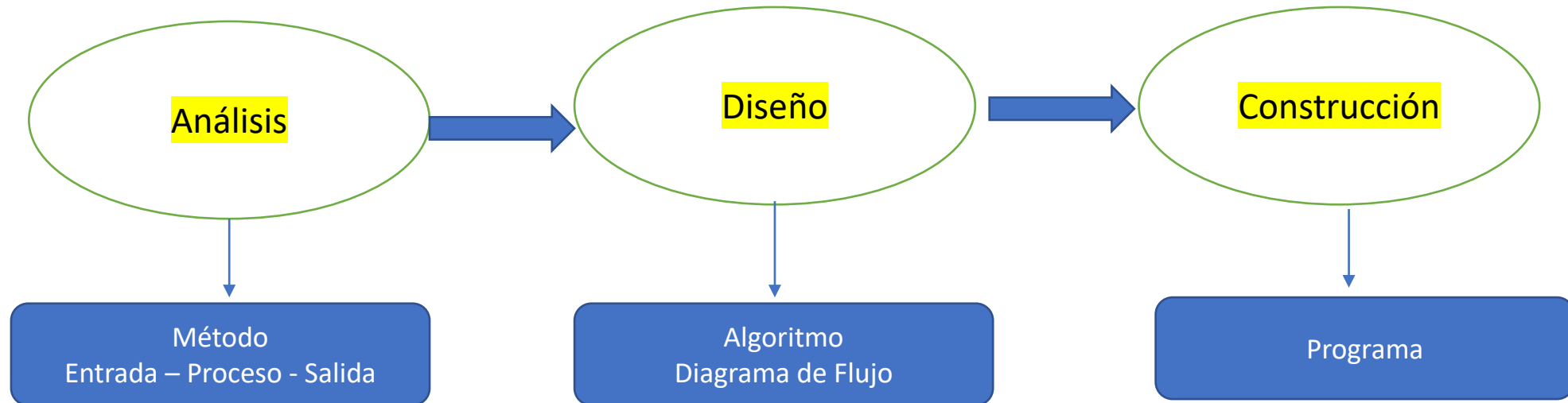




Iteración o ciclos controlados por condicional - WHILE



Metodología -> Pensamiento lógico estructurado





El futuro digital
es de todos

MinTIC



Iteración o ciclos controlados por condicional - WHILE



Análisis → Método Entrada-Proceso-Salida

Operaciones, cálculos, **estructuras de control**

Entrada
Se debe LEER

Proceso

Salida Se debe
IMPRIMIR

1

numero, inicializar contadores

2

Ciclo: Proceso del numero(PAR o IMPAR), WHILE
porque se repite mientras que numero no se -1

3

Condicional: Saber si es PAR o IMPAR ($\text{numero} \% 2$),
incrementar contadores

4

numero es Par o IMPAR

5

can_pares, can_impares
(CONTADORES, +1)



Iteración o ciclos controlados por condicional - WHILE

Algoritmo pares_impares

can_pares=0

can_impares=0

Escribir "Número entero: "

Leer num

Lectura del primero, para ingresar al ciclo while

Mientras num≠-1 **Hacer**

Si num%2=0 **Entonces**

Escribir num, " es PAR"

can_pares=can_pares+1

SiNo

Escribir num, " es IMPAR"

can_impares=can_impares+1

Fin Si

Escribir "Número entero: "

Leer num

Lectura del siguiente número para que no se quede en un ciclo sin fin.

Fin Mientras

Escribir "Cantidad de pares: ",can_pares

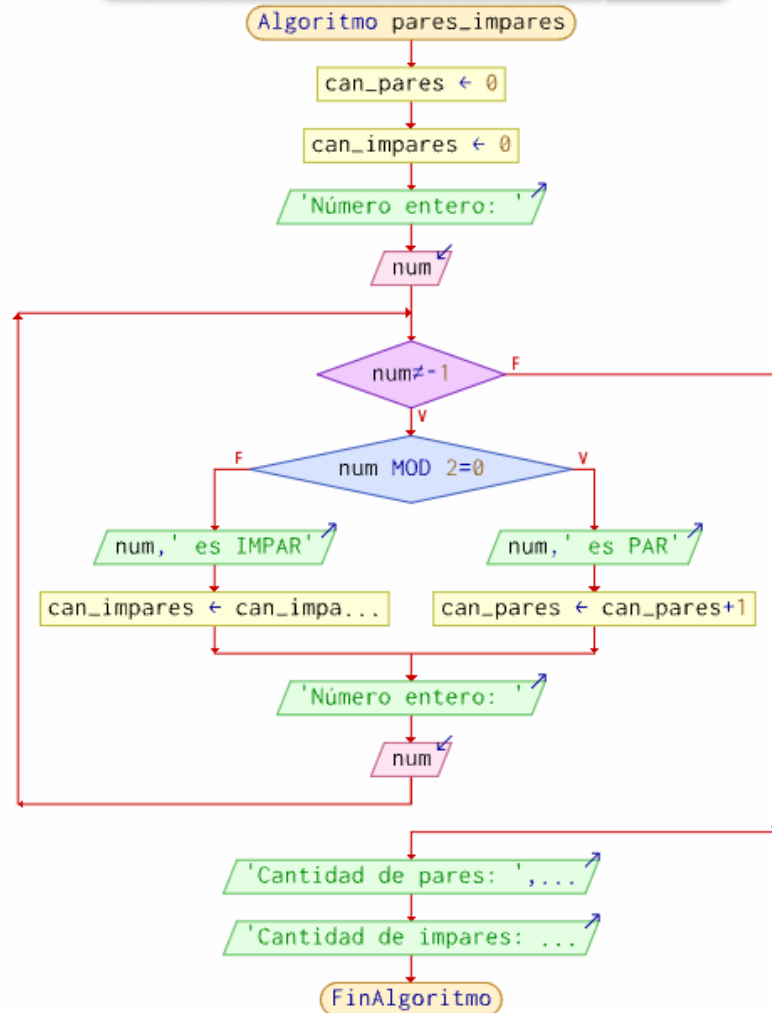
Escribir "Cantidad de impares: ",can_impares

FinAlgoritmo

Diseño → Algoritmo



Iteración o ciclos controlados por condicional - WHILE



Diseño → Diagrama de flujo



Iteración o ciclos controlados por condicional - WHILE



x
x
x

```
# Programa para hallar pares e impares
# Autor: Sergio Medina
# Fecha: 05/05/2022

can_pares=0
can_impares=0
num=int(input("Número entero: "))
while num!=-1:
    if num%2==0:
        print(num, " es PAR")
        can_pares+=1
    else:
        print(num, " es IMPAR")
        can_impares+=1
    num=int(input("Número entero: "))
print("Cantidad de pares: ",can_pares)
print("Cantidad de impares: ",can_impares)
```

Construcción → Programa

Estructuras de Control : Condicionales - Ciclos



El futuro digital
es de todos

MinTIC



Ejercicios Propuestos



El futuro digital
es de todos

MinTIC

● **Formulación y
Diagramación de soluciones**

● **Estructuras de Control**

»
Misión TIC 2022

xxx



**Misión
TIC 2022**