

SUBINDICADOR # 6

Diseña estructuras condicionales anidadas, empleando condicionales simples y compuestas en Python, para crear mejores soluciones algorítmicas.

1. Introducción:

El tema planteado tiene una importancia significativa ya que muestra al estudiante como este tipo de estructuras sistematizan las políticas o restricciones en los procesos más complejos de una organización.

MÓDULO 3: Valores booleanos, Instrucciones if-elif-else, Bucles while y for, Control de flujo, Operaciones lógicas y bit a bit, Listas y arreglos.

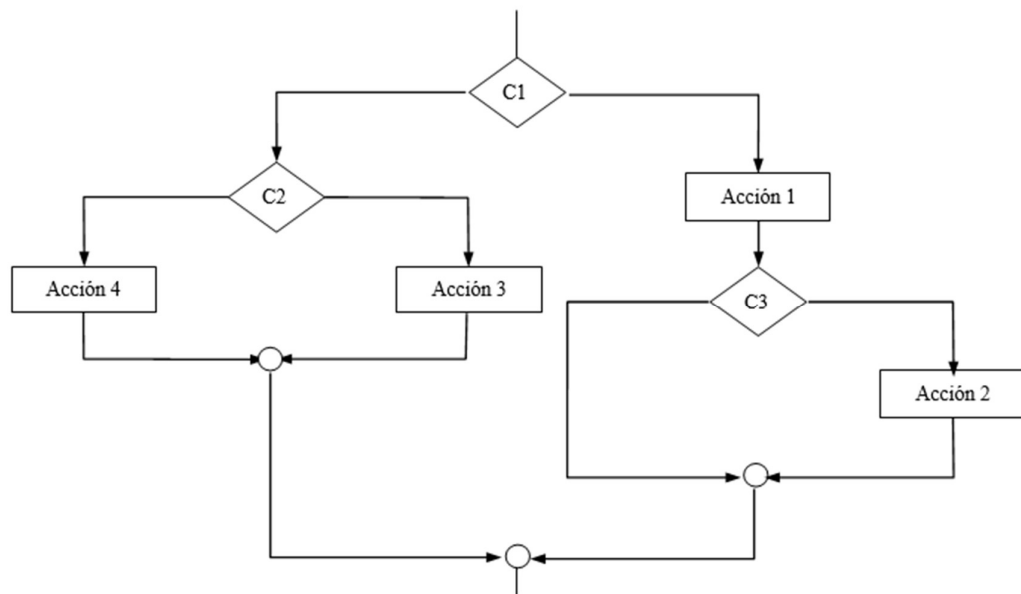
2. Condicionales anidadas.

a. Diseño de estructuras anidadas

Cuando tenemos que diseñar algoritmos que tienen diversas condiciones relacionadas, es decir evaluamos una misma variable para varios casos, debemos generar estructuras que denominan anidadas; ello implica la existencia de una condición dentro de otra.



Si se evalúan diversas variables, entonces las condiciones que se generan no deberán depender entre sí; una estructura condicional anidada se observa de la siguiente manera:



Se debe considerar que el diseño de la estructura no es el mismo siempre, varía dependiendo el proceso que se está resolviendo; si adjuntamos el pseudocódigo al diseño mostrado se muestra de la siguiente manera:

```

Si (C1) entonces
    Acción 1
    Si (C3) entonces
        Acción 2
    Fin_si
Sino
    Si (C2) entonces
        Acción 3
    Sino
        Acción 4
    Fin_si
Fin_si
  
```

Los operadores *and* y *or* también tienen su significado en las estructuras anidadas, cuando anidas hacia la derecha implica el uso de *and* y cuando se anida hacia la izquierda representa el uso de *or*.

También se debe contemplar una regla muy usada al momento de evaluar alternativas que deberán estar en estructuras anidadas, y consiste en diseñar N-1 condiciones ante la existencia de N alternativas; por ejemplo, si tenemos un cuadro como el mostrado, este tiene 4 alternativas por lo que solo será necesario anidar 3 condicionales (tener en cuenta que se asume que los datos ingresados son los correctos, es decir esta todo validado para que la regla se pueda aplicar).

AFP	% Descuento
UNION	14%
HORIZONTE	16%
INTEGRA	15%
PRIMA	14.5%

b. Declaración de if-else / elif en estructuras anidadas.

Cuando se estructura una serie de condicionales anidadas, una forma de poder reducir el código y evitar crear estructuras muy grandes de anidamiento es usar la forma simplificada de **if – else** que equivale a colocar **elif**, de ese modo reducimos considerablemente la apertura del código, observar el ejemplo:

```
print("Ingrese una Categoría:")
Cat=input()
if(Cat=="A"):
    print("Ventas")
else:
    if(Cat=="B"):
        print("Compras")
    else:
        if(Cat=="C"):
            print("MKT")
        else:
            print("Administración")
```

Ahora observar como se muestra el código traducido con **elif**

```
print("Ingrese una Categoría:")
Cat=input()
if(Cat=="A"):
    print("Ventas")
elif(Cat=="B"):
    print("Compras")
elif(Cat=="C"):
    print("MKT")
else:
    print("Administración")
```

c. Desarrollo de ejercicios tipo

Los ejercicios que veremos a continuación contienen la aplicación de todo tipo de condicionales, simples, compuestas y su aplicación en estructuras anidadas, en ese sentido las soluciones planteadas ya no tendrán el desarrollo del pseudocódigo.

1. Problema Prg1

Se ingresan 3 números **distintos**, desarrolle un algoritmo que permita hallar el mayor de ellos, para luego mostrarlo.

Solución:

a. Análisis.

¿Qué te piden que realices?

Encontrar el mayor de 3 números

¿Qué datos necesito conocer?

Los 3 números

b. Planteamiento Lógico.

Para poder obtener el mayor de tres números existen varias formas de analizar, una de ellas y la más común es evaluando al estilo de las matemáticas, donde se evalúa conjuntos de números, es decir evaluar de la siguiente manera:

(Num1>Num2>Num3)

Si (Num1>Num2) and (Num2>Num3) ...

Se concluye que el mayor es Num1, por lo tanto, se podría generar más preguntas para poder definir el número mayor, (en la solución usaremos otra variante.)

Otra forma de solucionar es creando una variable a la que se le asigne un valor y luego generar preguntas para poder condicionar si el número evaluado es menor al consecutivo y generar intercambios, esta forma puede ser muy útil cuando se ingresa N valores y tiene que hallar el mayor valor.

Mayor=Num1

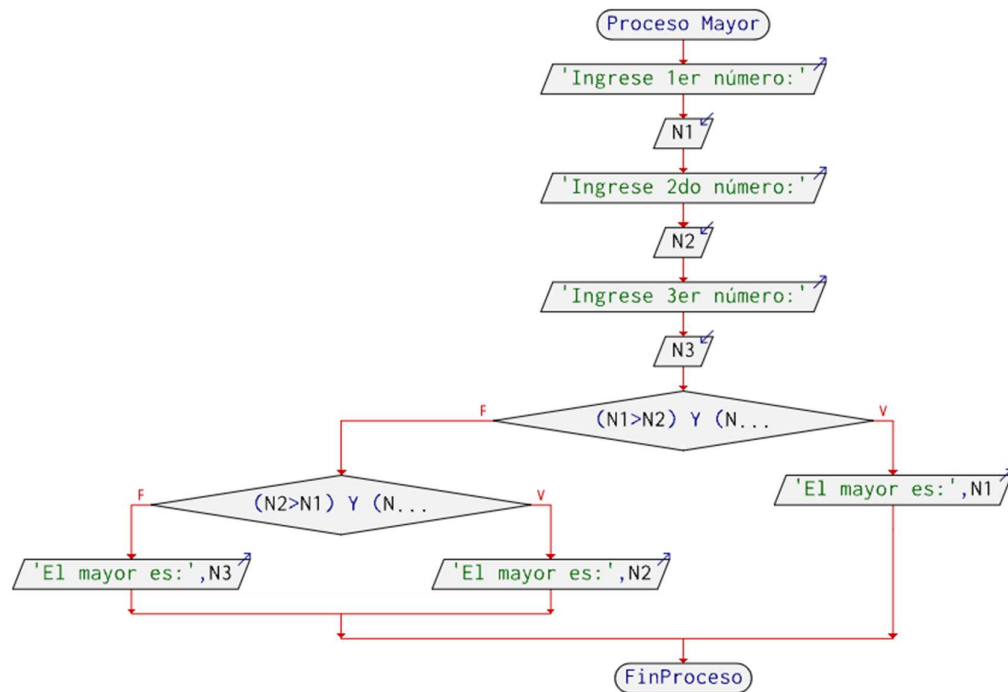
Si(Mayor<Num2) entonces

Mayor=Num2

De esa manera estaríamos ubicando al mayor de los números dentro de la variable denominada Mayor.

c. Diseño.

Diagrama de Flujo:



d. Lenguaje de Programación.

```

N1=int(input("Ing 1er #:"))
N2=int(input("Ing 2do #:"))
N3=int(input("Ing 3er #:"))
if (N1>N2) and (N1>N3) :
    print(f"El mayor es {N1}")
else:
    if (N2>N1) and (N2>N3) :
        print(f"El mayor es {N2}")
    else:
        print(f"El mayor es {N3}")
  
```

Esta forma de captura y mensajería es factible.

Al ejecutar el programa, se muestra lo siguiente:

```

Ing 1er #:90
Ing 2do #:500
Ing 3er #:3
El mayor es 500
  
```

Podemos variar el código y realizarlo usando elif, observe:

```
N1=int(input("Ing 1er #:"))
N2=int(input("Ing 2do #:"))
N3=int(input("Ing 3er #:"))
if(N1>N2)and(N1>N3):
    print(f"El mayor es {N1}")
elif(N2>N1)and(N2>N3):
    print(f"El mayor es {N2}")
else:
    print(f"El mayor es {N3}")
```

2. Problema Prg2

Diseñar un algoritmo que permita ingresar un número que representa a un mes del año, y luego indique de que estación del año se trata, ver cuadro:

Mes	Estación
1-3	Verano
4-6	Otoño
7-9	Invierno
10-12	Primavera

Solución:

a. Análisis.

¿Qué te piden que realices?

Indicar la estación del año

¿Qué datos necesito conocer?

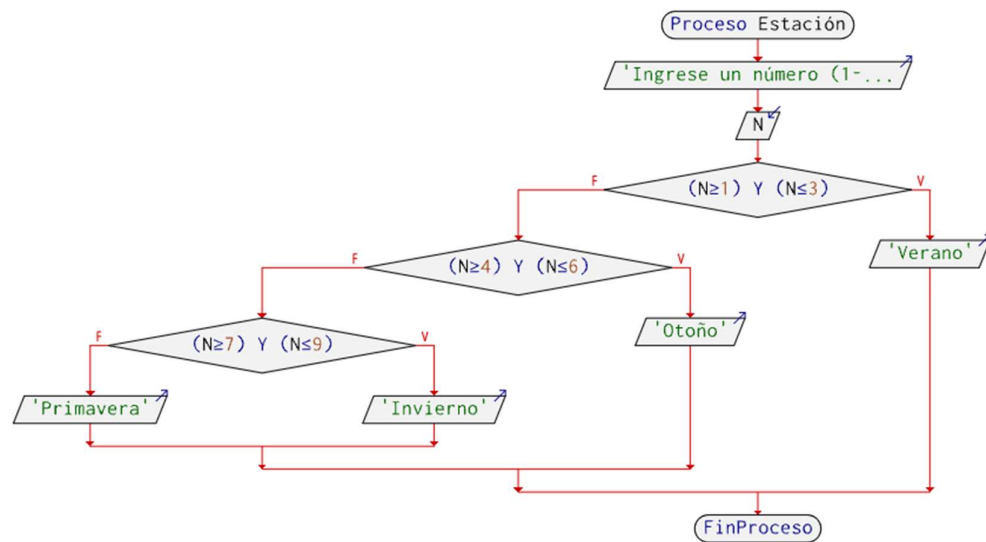
El número que representa al año

b. Planteamiento Lógico.

En este caso solo se requiere analizar rangos de valores en una estructura anidada, de modo que se pueda considerar los 4 casos mostrados, si consideramos que el ingreso de los números es el correcto, entonces aplicamos la fórmula de N-1 condicionales.

c. Diseño.

Diagrama de Flujo:



d. Lenguaje de Programación.

```

print("Ingrese un Número (1-12):")
N=int(input())
if (N>=1) and (N<=3) :
    print("Verano")
else:
    if (N>=4) and (N<=6) :
        print("Otoño")
    else:
        if (N>=7) and (N<=9) :
            print("Invierno")
        else:
            print("Primavera")
  
```

Se tenían 4 alternativas, se pregunta N-1 veces

Al ejecutar el programa, se muestra lo siguiente:

```

Ingrese un Número (1-12):
9
Invierno
  
```

Podemos variar el código y realizarlo usando elif, observe:

```

print("Ingrese un Número (1-12):")
N=int(input())
if (N>=1) and (N<=3) :
    print("Verano")
elif (N>=4) and (N<=6) :
    print("Otoño")
elif (N>=7) and (N<=9) :
    print("Invierno")
else:
    print("Primavera")

```

3. Problema Prg3

En una empresa el sueldo básico de los trabajadores es en base a la categoría que tienen, ver cuadro:

Categoría	Sueldo Básico
A	1100
B	2000
C	3500

Todos los trabajadores tienen una bonificación de acuerdo a cantidad de hijos que tienen, este se aplica al sueldo básico, ver cuadro:

Nro. Hijos	%
<2	0%
Entre 2 y 4	20%
>4	22%

Finalmente, los trabajadores tienen un descuento de acuerdo a la AFP en que se encuentren inscritos, este se aplica al total entre el sueldo básico + la bonificación por hijos:

AFP	%
PRIMA	12%
UNION	13%
HORIZONTE	12,7%
INTEGRA	13.9%

Realice un algoritmo que calcule y muestre el sueldo Neto a pagar, si este se obtiene de:

$$S_{\text{Neto}} = S_{\text{basico}} + \text{Bonificación} - \text{Descuentos}$$

Solución:

- a. Análisis.
¿Qué te piden que realices?

Calcular el sueldo neto

¿Qué datos necesito conocer?

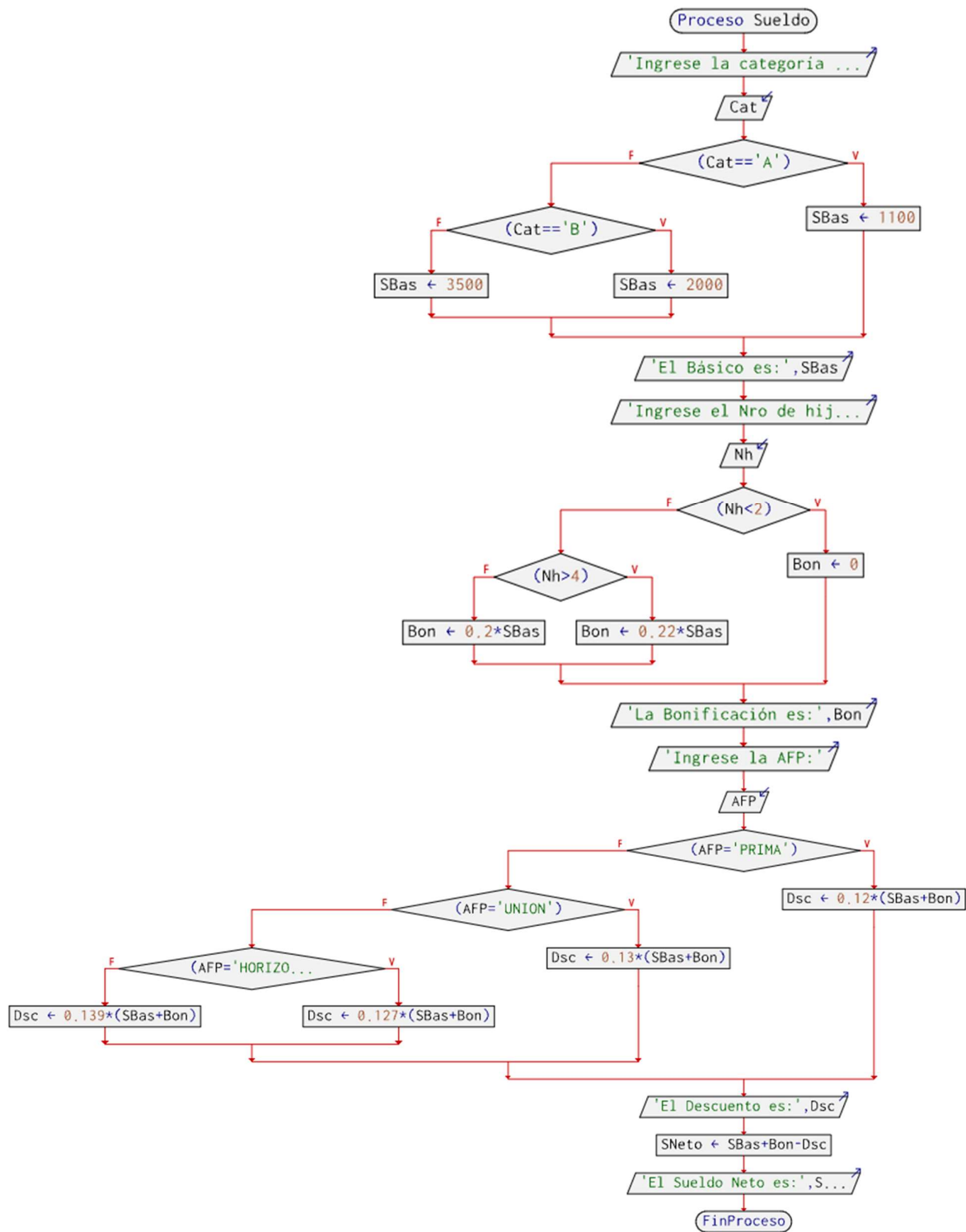
Categoría, Número de hijos y AFP

b. Planteamiento Lógico.

El desarrollo de este ejercicio no implica algún grado de complejidad, lo que si debe tener en cuenta es el orden en el que se debe operar cada requerimiento para poder conseguir el Sueldo Neto, para la solución vamos a romper un poco el esquema del orden de ingreso de datos, es decir vamos a ingresar los datos donde los requerimos, de modo que cambie un poco el concepto que tenemos de que todo se ingresa al principio del algoritmo.

c. Diseño.

Diagrama de Flujo:



d. Lenguaje de Programación.

```
print("Ingrese la categoría (A-B-C):")
cat=input()
if (cat=="A"):
    sbas = 1100
else:
    if (cat=="B"):
        sbas = 2000
    else:
        sbas = 3500
print("El Básico es:", sbas)
print("Ingrese el Nro de hijos:")
nh=int(input())
if (nh<2):
    bon = 0
else:
    if (nh>4):
        bon = 0.22*sbas
    else:
        bon = 0.2*sbas
print("La Bonificación es:", bon)
print("Ingrese la AFP:")
afp=input()
if (afp=="PRIMA"):
    dsc = 0.12*(sbas+bon)
else:
    if (afp=="UNION"):
        dsc = 0.13*(sbas+bon)
    else:
        if (afp=="HORIZONTE"):
            dsc = 0.127*(sbas+bon)
        else:
            dsc = 0.139*(sbas+bon)
print("El Descuento es:", dsc)
sneto=sbas+bon-dsc
print("El Sueldo Neto es:", sneto)
```

Observar que se está ingresando los datos antes de cada proceso.

Estamos usando variables para cada cálculo, y se está mostrando su valor.

Indicaba que se aplica el % a la suma del básico con la bonificación.

Al ejecutar el programa, se muestra lo siguiente:

```
Ingrese la categoría (A-B-C) :  
A  
El Básico es: 1100  
Ingrese el Nro de hijos:  
4  
La Bonificación es: 220.0  
Ingrese la AFP:  
PRIMA  
El Descuento es: 158.4  
El Sueldo Neto es: 1161.6
```

4. Problema Prg4

Se va ingresar un número positivo, si este resulta ser de 4 cifras, indique si es un número capicúa, ejemplos de números capicúas: 1221 / 3443.

Solución:

a. Análisis.

¿Qué te piden que realices?

Indicar si el valor ingresado es capicúa de 4 dígitos.

¿Qué datos necesito conocer?

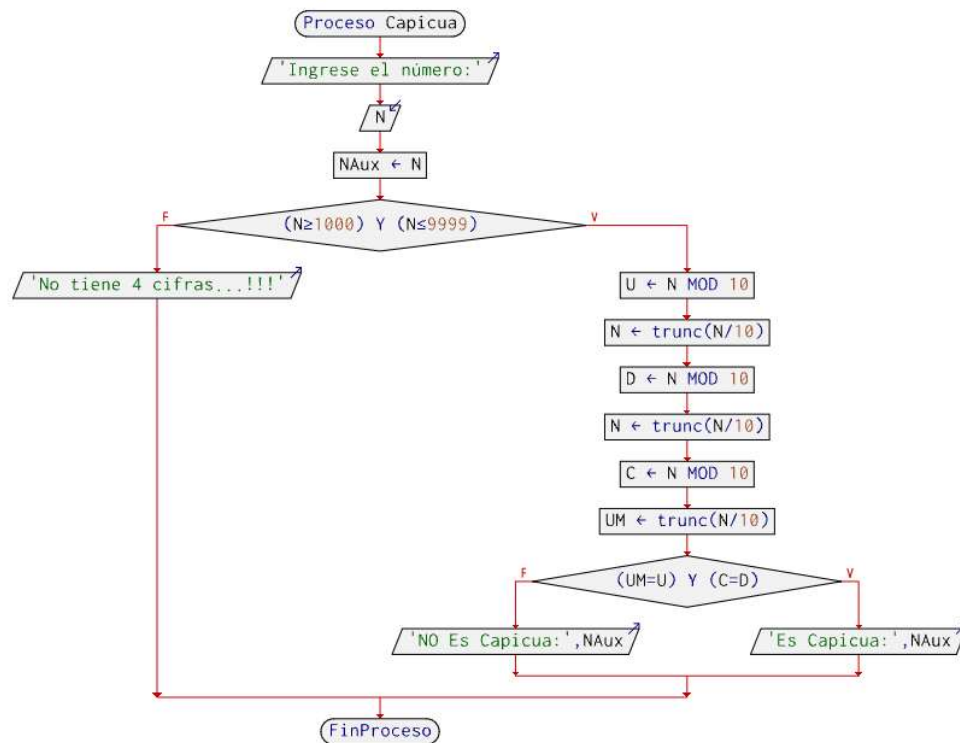
El número

b. Planteamiento Lógico.

Para desarrollar este ejercicio, debe primero identificar al número ingresado, es decir deberá tener 4 cifras, ya que esa es la condición inicial, si esta resulta cierta, ahora deberá evaluar si es un número capicúa (este término implica que el número se lee de derecha a izquierda igual como de izquierda a derecha); deberá descomponer el número para analizarlo.

c. Diseño.

Diagrama de Flujo:



d. Lenguaje de Programación.

```

print("Ingrese el número:")
n = int(input())
naux = n
if (n >= 1000) and (n <= 9999):
    u = n % 10
    n = n // 10
    d = n % 10
    n = n // 10
    c = n % 10
    um = n // 10
    if (um == u) and (c == d):
        print("Es Capicua:", naux)
    else:
        print("NO Es Capicua:", naux)
else:
    print("No tiene 4 cifras...!!!")
  
```

Descomposición total del
número de 4 cifras.

Al ejecutar el programa, se muestra lo siguiente:

```

Ingrese el número:
4554
Es Capicua: 4554
  
```

Ahora podemos generar variaciones a este código, ya dependerá del grado de práctica y desarrollo que se tiene al momento el mejorarlo y comprenderlo, observe:

```
print("Ingrese el número:")
n = int(input())
if (n>=1000) and (n<=9999):
    if ((n//10)//10)//10==n%10 and ((n//10)//10)%10==(n//10)%10:
        print("Es Capicua:",n)
    else:
        print("NO Es Capicua:",n)
else:
    print("No tiene 4 cifras...!!!")
```

5. Problema Prg5

Desarrolle un algoritmo que permita entregar un monto a diversas personas, todo dependerá de la edad y del tipo de sangre, de acuerdo al cuadro mostrado.

Edad	Tipo de Sangre	Monto entregado
18	A	3000
19..35	A	1500
	B	2500
36..55	AB o A	4000
Otros casos	Otros casos	1800

Solución:

a. Análisis.

¿Qué te piden que realices?

Entregar un monto de dinero

¿Qué datos necesito conocer?

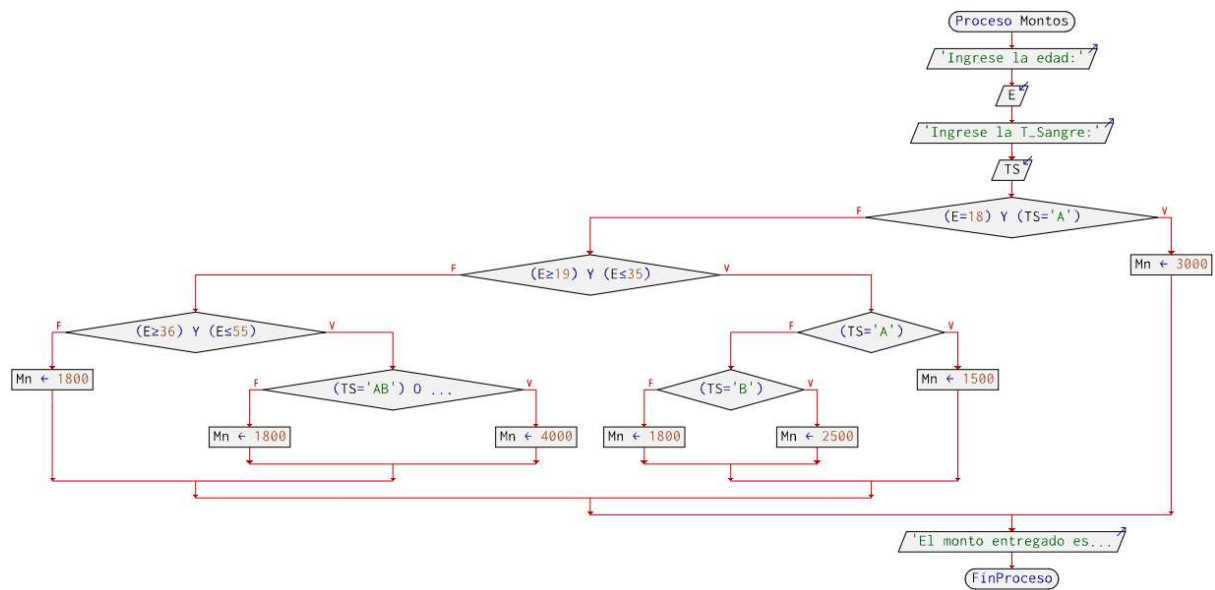
Edad y tipo de sangre.

b. Planteamiento Lógico.

Para desarrollar este ejercicio debe considerar el cuadro de doble entrada mostrado, de modo que pueda generar las condiciones adecuadas y obtenga los montos adecuados. Tenga cuidado al momento de interpretar los datos, sobre todo con otros casos, ya que deberá analizar donde deberá colocar el valor indicado.

c. Diseño.

Diagrama de Flujo:



Observe detenidamente que está ocurriendo con este punto indicado en el cuadro:

Otros casos	Otros casos	1800
-------------	-------------	------

Si observa el gráfico detenidamente podrá ubicarlos en diversos lugares, es decir se está distribuyendo el monto de 1800 de tal manera que no existan condicionales que terminen por indicar que no existe un monto que entregar; dice claramente en otros casos deberá entregarse 1800.

d. Lenguaje de Programación.

```
print("Ingrese la edad:")
e = int(input())
print("Ingrese la T_Sangre:")
ts = input()
if (e==18) and (ts=="A"):
    mn = 3000
else:
    if (e>=19) and (e<=35):
        if (ts=="A"):
            mn = 1500
        else:
            if (ts=="B"):
                mn = 2500
            else:
                mn = 1800
    else:
        if (e>=36) and (e<=55):
            if (ts=="AB") or (ts=="A"):
                mn = 4000
            else:
                mn = 1800
        else:
            mn = 1800
print("El monto entregado es:",mn)
```

Al ejecutar el programa, se muestra lo siguiente:

```
Ingrese la edad:
30
Ingrese la T_Sangre:
B
El monto entregado es: 2500
```

3. Actividad

A) Ejercicios en general

Desarrolle los siguientes ejercicios, considerando lo desarrollado anteriormente.

- a) El sueldo de los empleados depende del área de trabajo (ver cuadro)

Área	Sueldo
Ventas	750
MKT	900
Almacén	1000
Otros	1200

Determine cuánto gana (S_Neto) un empleado si se sabe que al básico debemos agregarle un monto adicional de acuerdo al nro de hijos que tenga, ver cuadro:

Nro de Hijos	% adicional aplicado al Básico
0	0
1 a 3	15%
4 a más	20%

$$S_Neto = \text{Sueldo} + \% \text{ adicional por hijos}$$

- b) En una tienda de repuestos los costos de las lunas son como siguen:

Auto	Costo de la luna
Toyota	150
Nissan	200
Hyundai	100

Si la compra las lunas de Toyota en una cantidad mayor a 2 tiene un descuento del 20% sobre el costo, si compra las lunas de Nissan en una cantidad mayor a 3 tiene un descuento del 25% sobre el costo y si es de Hyundai se aplica directo un 10% de descuento. Realice un algoritmo que muestre cuanto pagará la persona por comprar estas lunas finalmente (incluir el descuento).

Deberá ingresar la marca del auto y la cantidad de lunas a comprar.

- c) Desarrolle un algoritmo que permita ingresar 3 números luego sume el primero con el segundo y el segundo con el tercero, verifique si los valores que se generaron fueron diferentes, de ser cierto ello ingrese otro número y agréguele (sumar) a los números generados, en caso contrario ingrese otro número y multiplíquelo a los dos anteriores. Mostrar los números que se generaron en ambos casos.
- d) En un hotel los costos por día de las habitaciones son de acuerdo al tipo, según la tabla:

Tipo	Costo
A	75
B	65
C	55
D	40

Tener en cuenta que si la habitación es de tipo A o B y cuenta con jacuzzi se le debe aumentar al costo s/. 5

Además, dependiendo el número de días que se pueda quedar se le otorga un descuento sobre el pago diario a realizar, según la tabla:

Días	Descuento
1..3	2%
4..7	10%
8 a más	25%

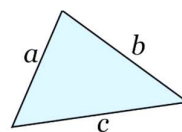
Realizar un algoritmo en donde se ingrese el tipo de habitación y el número de días que se quedará en el hotel, de modo que se pueda calcular y mostrar el total a pagar por la persona al momento de que se retire del hotel.

$$\text{Total a pagar} = \text{Costo por día} * \text{número de días}$$

- e) En un bar el costo de los tragos son como siguen: Ron – 20 soles, Cerveza – 10 soles y Vino – 25 soles; el bar otorga descuentos sobre el precio teniendo en cuenta la cantidad de bebidas a comprar: si el número de Rones a comprar sobrepasa los 5 entonces al precio se le quita 2 soles en caso contrario se le quitará 1 sol; si el número de Cervezas a comprar sobrepasa los 20 entonces al precio se le quita 3 soles en caso contrario no se le quita nada y si el número de Vinos a comprar sobrepasa los 4 entonces al precio se le quita 7 soles. Determinar cuánto pagará el cliente que compra estas bebidas en el bar.

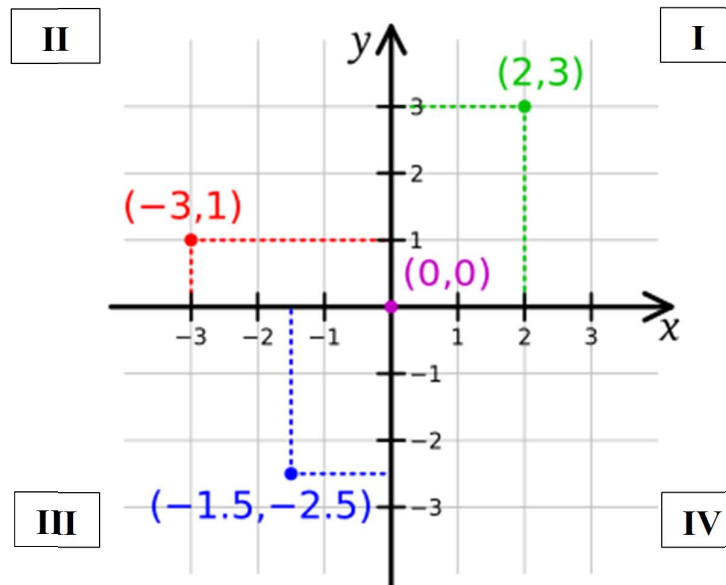
$$\text{Total a pagar} = \text{Precio (con descuentos)} * \text{Cantidad}$$

- f) Desarrolle un algoritmo que pida la longitud de los lados de un triángulo, luego indique con mensajes si era equilátero, isósceles o escaleno. Un triángulo es equilátero si sus tres lados son iguales, isósceles si dos lados son iguales y escaleno si sus 3 lados son distintos; para este caso adicionalmente deberá validar la existencia del triángulo previamente; para ello considerar el cumplimiento de lo siguiente:



$$\begin{aligned} a + b &> c \\ b + c &> a \\ c + a &> b \end{aligned}$$

- g) Desarrolle un algoritmo que le solicite ingresar las coordenadas cartesianas de un punto (X e Y), luego indique en que cuadrante se encuentra dicho punto o si se trata del origen de coordenadas.



- h) Desarrolle un algoritmo que permita ingresar un número que representa a un día del mes, otro número que representa al mes y el año al que pertenece; deberá mostrar la fecha completa: Ejemplo:

Si se ingresa: Día = 16; Mes = 6; Año = 2020
Deberá salir el mensaje: **16 de junio del 2020**

Deberá considerar:

- Si el día está mal ingresado, ya no puede continuar el programa.
- Si el mes está mal ingresado, ya no puede continuar el programa.
- Además, aparte de indicar el año, deberá solicitar indicar si este es bisiesto; recuerde que en estos años Febrero tiene 29 días.
- Recordar que existen meses con 30 y 31 días.

4. Fuentes consultadas:

- A) <https://edube.org/learn/programming-essentials-in-python-part-1-spanish>
B) Edison Zavaleta C. (2005). *Fundamentos de Programación*. Perú, Editorial Abaco-Lima.