



El futuro digital
es de todos

MinTIC

Universidad
Industrial de
Santander



Mision
TIC2022

ESTRUCTURAS DE CONTROL CONDICIONAL O SELECTIVAS



INTRODUCCIÓN

Al momento de construir una instrucción de tipo condicional se debe comprender que dichas instrucciones están diseñadas con el objetivo de ayudar en la toma de decisiones; por tal razón, se necesita diseñar un algoritmo que permita visualizar la variedad de opciones que se pueden generar. Estas estructuras de control condicional utilizadas en programación, son las que permiten evaluar un sinnúmero de elecciones que están alineadas a un conjunto inicial de condiciones; si es verdadera realiza una acción, y si es falsa realiza otra.

Existen varios tipos de instrucciones de tipo condicional; algunas pueden ser simples y otras pueden ser múltiples; sin embargo, solo se generan dos resultados, los cuales son verdadero o falso; en el caso del lenguaje de programación Python se utilizan los términos en inglés True o False.

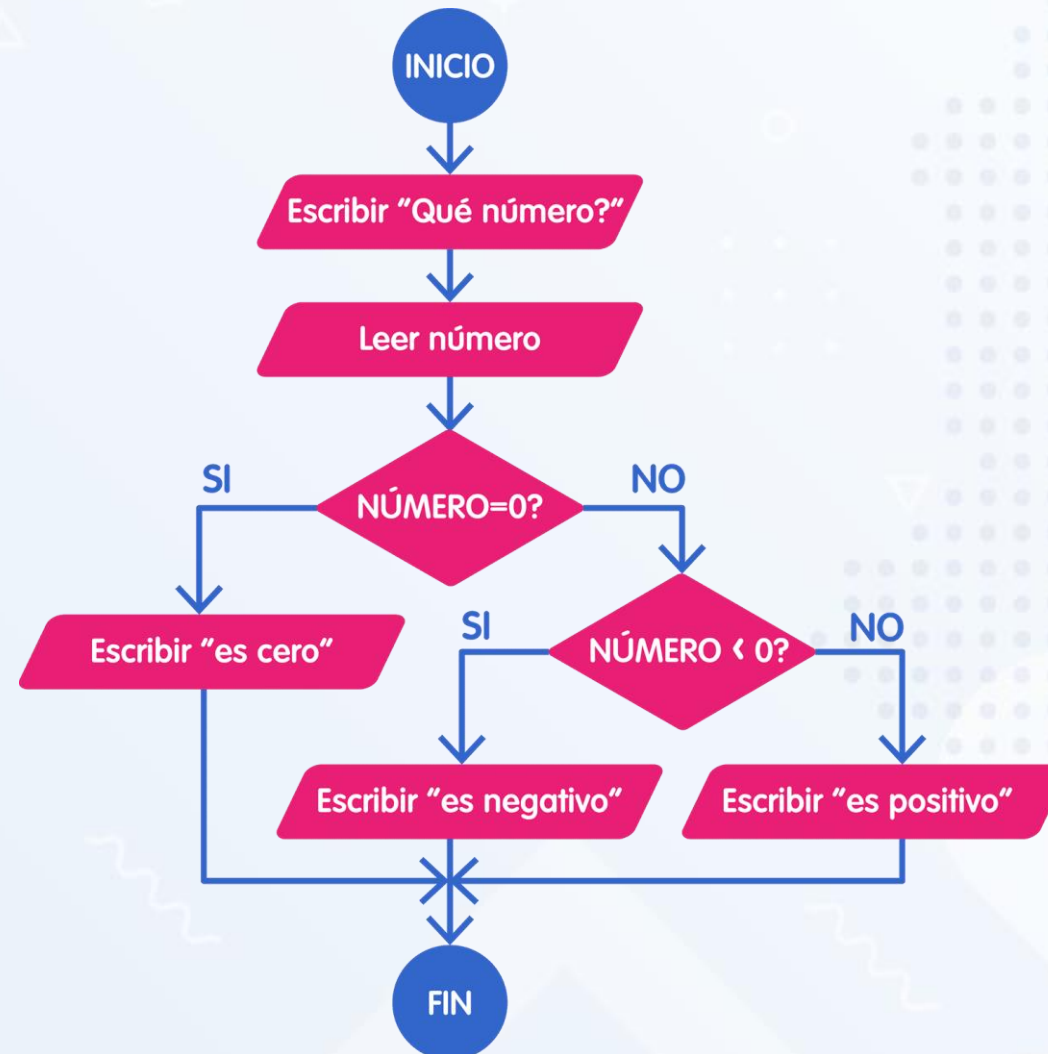
De esta manera, para evaluar la acción a realizar se usan comandos que representan los operadores de comparación vistos en el eje temático anterior. Los resultados de cada análisis comparativo se expresan en operadores lógicos, tal y como se ven en la siguiente tabla:

Símbolo Utilizado	Significado del Símbolo	Ejemplo	Resultado Obtenido
==	Igual que	2 == 7	FALSO
!=	Diferente que	Guitarra = piano	VERDADERO
<	Menor Que	3 < 22	VERDADERO
>	Mayor Que	1 > 16	FALSO
<=	Menos o Igual Que	28 <= 28	VERDADERO
>=	Mayor o Igual Que	2 >=3	FALSO

De igual forma se pueden tratar las condiciones múltiples, pero en este caso se tiene que enlazar con operadores lógicos como lo son el "y" y el "o", más comúnmente usados en programación con sus equivalentes en el idioma inglés "and" y "or", tal y como se observa en la siguiente tabla:

Operador	Ejemplo	Solución	Resultado
And (y)	5==8 and 7 < 12	0 y 0	FALSO
	1 < 10 and 8 > 7	1 y 1	VERDADERO
	3 > 1 and 3 < 1	1 y 0	FALSO
	6 < 2 and 6 > 11	0 y 1	FALSO
Or (o)	3 == 3 or 15 < 3	1 y 0	VERDADERO
	3 < 1 or 12>2	0 y 1	VERDADERO
	4>10 or 1 == 7	0 y 0	FALSO
	5 == 5 or 3 > 1	1 y 1	VERDADERO
Xor (o excluyente)	4 == 4 xor 9 > 5	1 o 1	FALSO
	4 == xor 9 < 5	1 o 0	VERDADERO

En el siguiente diagrama de flujo se aprecia el uso de las condiciones para solucionar el problema de saber si un número ingresado es positivo o negativo.



Material de estudio complementario

Estructuras condicionales en Python

Referencias Bibliográficas

LUTZ, Mark. Learning Python. O'Reilly Media: 5° Edición, 2013. 1600 p. ISBN: 978-1449355739

GUZDIAL, Mark y ERICSON, Bárbara. Introducción a la computación y programación con Python: Un enfoque multimedia. México: Pearson Educación, 2013. ISBN: 978-607-322-049-1

RASCHKA, Sebastián. Python Machine Learning. Packt Publishing, 2015. 454 p. ISBN: 978-1783555130



El futuro digital
es de todos

MinTIC

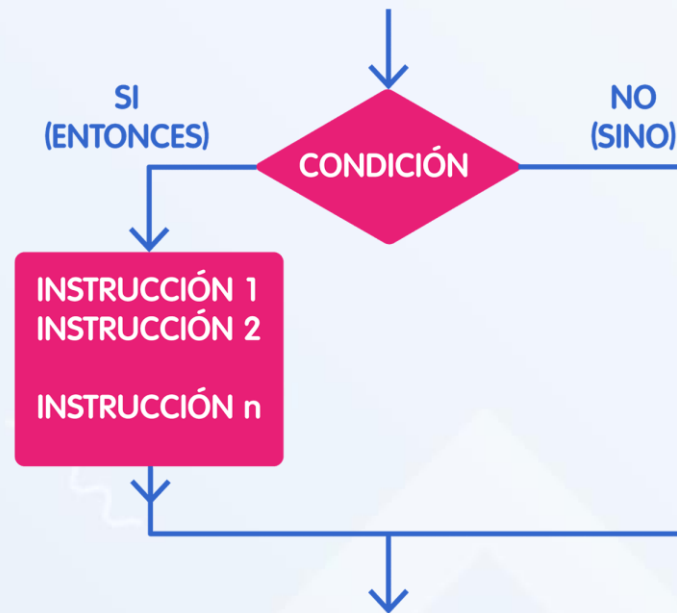
CONTROL DE FLUJO CONDICIONAL (IF, ELSE, ELIF)



Para construir líneas de comandos condicionales en el lenguaje Python, se pueden utilizar tres órdenes concretas: if, else y elif.

2.1. El comando IF

El comando if (si) es utilizado para plantear una condición. Si esta condición es evaluada como verdadera, se ejecuta una acción establecida mediante un código de instrucción o instrucciones escritas inmediatamente en las líneas siguientes a dicha condición. En el siguiente diagrama de flujo se aprecia la secuencia de un algoritmo condicionado.



De igual manera se puede ejemplarizar con el siguiente código en Python.

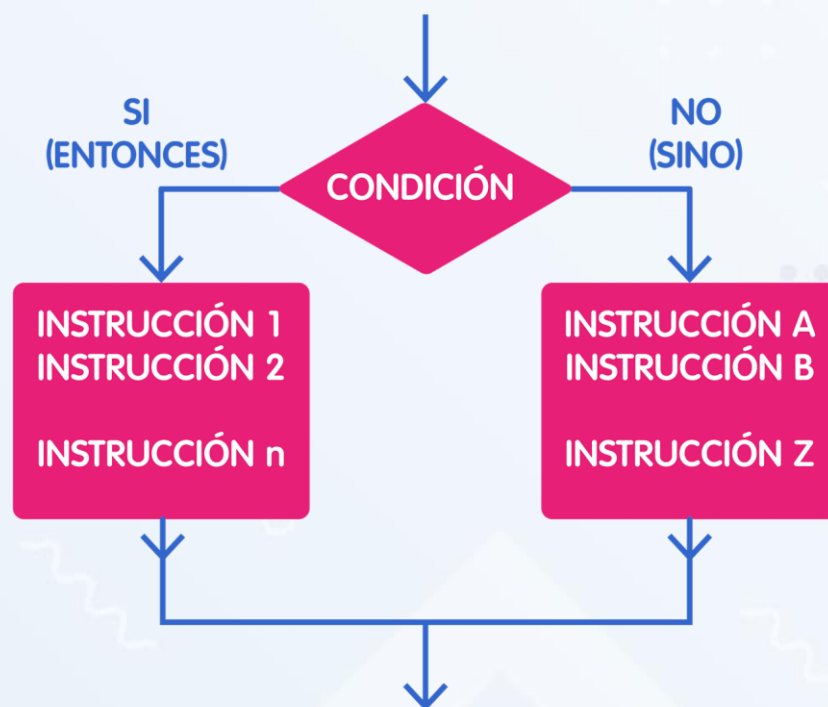
```
num = input('Introduzca un número: ')\nIf num == 100:\n    print("Usted escribió el 100")
```

Es recomendado el manejo de la tabulación pues gracias a ella se pueden establecer las acciones que dependen estrictamente de la condición planteada. En otras palabras, con el si condicional se establece la condición y para poder indicar el inicio de las acciones a realizar se deben utilizar los dos puntos (:). El si condicional es evaluado y en el caso de que sea verdadero se ejecutan todas aquellas acciones asociadas a él. En el caso de que la condición no se cumpla no se realiza ninguna acción.

2.2. El comando ELSE

Otro comando utilizado junto a la instrucción if o si condicional, es el comando else conocido en español como el "si no". Este comando es el que permite relacionar las acciones que se deberían realizar en el caso de que la condición sea evaluada como falsa.

En el siguiente diagrama de flujo se aprecia la secuencia de un algoritmo condicionado con dos opciones posibles, es decir, en caso de verdadero realiza unas instrucciones y en caso de falso realiza otras instrucciones.

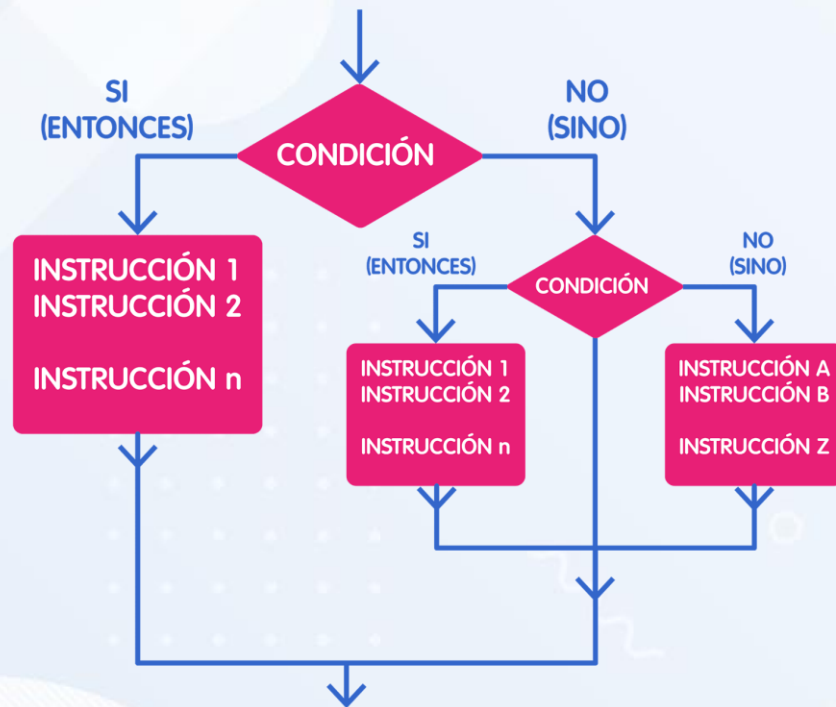


El siguiente ejemplo permite identificar los comandos utilizados en Python.

```
num = input('Introduzca un número: ')  
If num == 100:  
    print("Usted escribió el 100")  
else:  
    print("Usted no escribió el 100")
```

2.3. El comando ELIF

También es posible establecer códigos condicionales anidados; para realizar esta acción, se utiliza la última orden elif, que significa "si no, si", y permite concatenar condiciones. En el siguiente diagrama de flujo se aprecia la secuencia de un algoritmo condicionado.



Con el siguiente ejemplo se aprecia la sintaxis utilizada en Python.

```
num = input('Introduzca un número: ')
If num == 100:
    print("Usted escribió el 100")
elif If num > 100:
    print("Usted escribió un número mayor a
100")
else:
    print("Usted escribió un número menor a
100")
```


Material de estudio complementario

Estructuras condicionales en Python

Referencias Bibliográficas

LUTZ, Mark. Learning Python. O'Reilly Media: 5º Edición, 2013. 1600 p. ISBN: 978-1449355739

GUZDIAL, Mark y ERICSON, Bárbara. Introducción a la computación y programación con Python: Un enfoque multimedia. México: Pearson Educación, 2013. ISBN: 978-607-322-049-1

RASCHKA, Sebastián. Python Machine Learning. Packt Publishing, 2015. 454 p. ISBN: 978-1783555130



El futuro digital
es de todos

MinTIC

CONDICIONALES MÚLTIPLES



Como se visualizó en la sección anterior, cuando se presentan situaciones con más de una condición que dependen unas de otras, estas se pueden tratar mediante el manejo de varias sentencias o comandos if, o mediante el manejo adecuado del comando elif. Sin embargo, en muchos casos cuando se hacen necesarias múltiples condiciones, este tipo de programación necesita mayor cantidad de líneas de comandos.

Es allí donde se hace necesario el uso de operadores lógicos tales como el and y el or, los cuales proporcionan generalmente formas útiles de simplificar las sentencias condicionales anidadas.

Como ejemplo de lo anterior se visualiza el siguiente código que utiliza dos if anidados:

```
if 0 < x:  
    if x < 10:  
        print "x es un número positivo de un solo dígito."
```

En este caso la sentencia print sólo se llegaría a ejecutar si se llegan a superar las dos condiciones. Este ejemplo puede ser totalmente reemplazado con una sola línea de código usando el operador and:

```
if 0 < x and x < 10:  
    print "x es un número positivo de un solo dígito."
```

Este tipo de condiciones son comunes, por lo que Python nos proporciona una sintaxis alternativa similar a la notación matemática:

```
if 0 < x < 10:  
    print "x es un número positivo de un solo dígito."
```

Semánticamente esta **condición es la misma que la expresión booleana compuesta y que la condicional anidada.**

Material de estudio complementario

Estructuras condicionales en Python

Referencias Bibliográficas

LUTZ, Mark. Learning Python. O'Reilly Media: 5º Edición, 2013. 1600 p. ISBN: 978-1449355739

GUZDIAL, Mark y ERICSON, Bárbara. Introducción a la computación y programación con Python: Un enfoque multimedia. México: Pearson Educación, 2013. ISBN: 978-607-322-049-1

RASCHKA, Sebastián. Python Machine Learning. Packt Publishing, 2015. 454 p. ISBN: 978-1783555130