

NOMBRE Ejercicios de programación
INSTITUCIÓN _____
MATERIA _____

FECHA _____
PROFESOR _____
CURSO _____
NOTA _____

1.) Solo instrucciones secuenciales:

a.) Lea dos números enteros e imprima su suma en pantalla. ✓

Propuesta de solución:

```
num1 = int(input())
```

```
num2 = int(input())
```

```
print(num1 + num2)
```

¿Cómo sé que este programa funciona? (i.e. Suma dos números enteros)

- 1.) Selecciono dos números enteros cualesquiera para probar el programa - ^{Ambos} Diferentes a 0, idealmente.

2.) Miro si el resultado impreso en pantalla es la suma de los dos números dados.

$$4 + 0 = 4 \quad 4 + 1 = 5$$

$$4 - 0 = 4 \quad 4 - 1 = 3$$

3.) Si quiero asegurarme más de que el programa, en efecto, suma dos números enteros, entonces lo pruebo con otros dos números enteros diferentes...

Arch. Datos Entradas

Arch. Datos Salida

3

4

1

\uparrow
num1 = random(-100, 100, 'cerrado') } Gen.
num2 = random(-100, 100, 'cerrado') } Datos Entrada.

2.) Considerar los casos:

Lea dos números enteros, y diga si la suma de éstos es un número positivo, negativo o cero. ✓

Propuesta de solución:

```
num1 = int(input())
```

```
num2 = int(input())
```

```
if (num1 + num2 > 0):
```

```
    print('La suma de', num1, 'y', num2, 'es un número positivo')
```

```
elif (num1 + num2 < 0):
```

```
    print('La suma de', num1, 'y', num2, 'es un número negativo')
```

```
else:
```

```
    print('La suma de', num1, 'y', num2, 'es cero')
```

¿Cómo sé que este programa funciona?

1.) En este caso, no puedo simplemente seleccionar dos números enteros cualquiera, sino que debería revisar que par de números enteros cumplen cada una de las 3 condiciones que muestran mensajes en pantalla, y probar al menos un par de números por cada condición.

¿Cómo se cumple cada condición?

E.g.: $\begin{array}{r} 3 \\ 0 \end{array} \rightarrow$ La suma de 3 y 0 es un número positivo.

$\begin{array}{r} -3 \\ 0 \end{array} \rightarrow$ La suma de -3 y 0 es un número negativo.

$\begin{array}{r} 3 \\ -3 \end{array} \rightarrow$ La suma de 3 y -3 es cero.

Con estos tres casos, valido que los dos números seleccionados se suman, y según el resultado de la suma, así es el mensaje.

No obstante, aquí puede que me comience probar los 'valores extremos'; para convencerme más de que el programa es correcto, así:

Si $a, b \in \mathbb{Z}$, ent:

-1 : Núm. negativo ^{entero} más cercano a cero.

0 : Cero.

1 : Núm. positivo ^{entero} más cercano a cero.

Entonces, probar dos números a, b , tales que

$$a + b = 0 \Rightarrow 1 + (-1) = 0$$

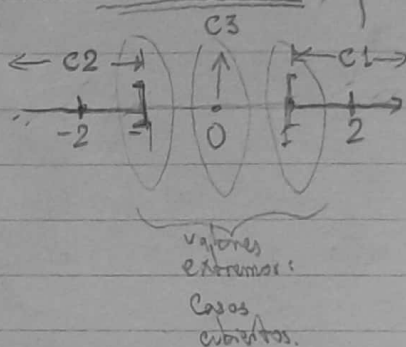
$$a + b = 1 \rightarrow 0 + 1 = 1$$

$$a + b = -1 \rightarrow -1 + 0 = -1$$

$\begin{array}{r} 1 \\ -1 \end{array} \rightarrow$ La suma de 1 y -1 es cero.

$\begin{array}{r} 0 \\ 1 \end{array} \rightarrow$ La suma de 0 y 1 es un número positivo.

$\begin{array}{r} 1 \\ -1 \end{array} \rightarrow$ La suma de -1 y 0 es un número negativo.



\Rightarrow De forma similar al ejercicio 1, puedo convencerme más si dan los pruebas con otros pares de números...

Entonces, ^{CEval.} Variable
 $\text{num1} = \text{rand}[-100, 100]$
 $\text{num2} = \text{rand}[-100, 100]$

n casos de prueba

(Para cualquier cond; pero valido que la op sea una suma y da salida a val.)

$\text{num1} = 0$
 $\text{num2} = 1$

1 caso de prueba
 Para condición 1
 (Valor extremo)

$\text{num1} = -1$
 $\text{num2} = 0$

1 caso de prueba
 Para condición 2
 (Valor extremo)

$\text{num1} = 1$
 $\text{num2} = -1$

1 caso de prueba
 Para condición 3
 (Valor extremo)

• py \Rightarrow ¿Cuántos casos de evaluación quiero? (+) ¿Cómo los quiero generar? : Sean de las lecturas (por variante)

Hay que emularlo con un while + break → $\left\{ \begin{array}{l} \text{while True:} \\ \quad \# \text{ Proc} \\ \quad \# \text{ Actualizar condición} \\ \quad \text{if (condition):} \\ \quad \quad \text{break} \end{array} \right.$

Python No tiene un constructo [do-while loop]

3.) Ciclos no anidados.

Tres variantes: MQ, Haga Hasta, Para

- a.) Lea un número entero n . Luego, lea n números enteros, e imprima su suma. → for
no negativo (mayor o igual a 1)
- b.) Lea un número entero n . Si este número no es 0, siga leyendo números hasta que este sea 0. Al final, imprima la suma de los números leídos. → while
no negativo (mayor o igual a 0)

Propuestas de solución:

a.) $n = \text{int}(\text{input}())$, $\text{sum} = 0$
 for (i in range(1, n+1, 1)):
 $\text{sum} = \text{sum} + \text{int}(\text{input}())$
 print(sum)

b.) $n = \text{int}(\text{input}())$, $\text{sum} = n$
 while ($n \neq 0$):
 $n = \text{int}(\text{input}())$
 $\text{sum} = \text{sum} + n$
 print(sum)

- c.) Lea dos números enteros m y n , e imprima en pantalla el resultado de sumar n veces el número m con n (Asuma que el programa siempre recibirá un valor $n > 0$).

Propuesta de solución:

Propuesta de solución INCORRECTA

c.) $m = \text{int}(\text{input}())$
 $n = \text{int}(\text{input}())$, $\text{sum} = m$
 for (i in range(1, n+1, 1)):
 $\text{sum} = \text{sum} + m$
 print(sum)

$m = \text{int}(\text{input}())$
 $n = \text{int}(\text{input}())$
 print($m * n$)

⊗

¿Cómo pruebo que las propuestas de solución en 3a, 3b y 3c hacen lo que dice ser respectivas enunciadas?

3a). Primero, debo revisar, en este caso, que se lean $n+1$ números enteros (Uno para leer n ; y los demás que son los números enteros a sumar — n números enteros a sumar). Luego, se debe evaluar que la suma de estos números se imprima en pantalla. Además, como es usual, se deberán considerar 'valores extremos'. En este caso, sería 1, porque este es el número entero más cercano a un número entero que no es mayor o igual a 1.

1	4	2	9	4	5
4		6		1	
		3		2	
				1	
				1	

Valor extremo

↓

1 C. Bral Fijo

n C. Bral / Valor n : aleatorio: $[1; 10] \rightarrow > 0$
 aleatorios n valores: aleatorios: $[-10; 10] \rightarrow \in \mathbb{Z}$

Lecturas
Consecutivas de cada lectura

3b.) Primero, debo verificar que se lee 1 número; luego, que se lean números hasta que el último número ingresado sea cero. (En este caso, note que, al igual que en el caso 3a, siempre se lee al menos 1 número entero (pero este puede ser < 1 en este caso) y la lectura de otros números es similar (pero en este caso, el último número leído siempre es 0; si el primer número es 0, no se leen más números). El resto de la prueba es similar al caso 3a.

Valores extremos o claves a considerar aquí.

1) El primer número es 0. \Rightarrow No se leen números \Rightarrow No hay nada que mostrar. \Rightarrow caso particular valor extremo!

2) Se lee un número diferente de 0, y luego, un 0 \Rightarrow suma = el número dif. de 0. \Rightarrow caso particular

3) Se lee un número diferente de 0, y luego, otro dif. de 0, y un 0 \Rightarrow suma = (la suma de los números leídos)

Los demás casos deberían ser para validar la suma entre los números leídos.

0	2	2	2	6
	0		4	
			0	

Valores extremos o
Casos especiales

Datos a leer fijos

n : # de datos a leer: aleatorio: $[1; 10]$

$n-1$ datos a leer: aleatorio: $[-10; 10]$

1 dato a leer: siempre 0 \rightarrow fijo

3c.)