



503203/503201/503215 Programación Programación Nivel Certamen

Javier Vidal

28 de abril de 2022

- 1.- ADDITION es un computador experimental que sólo puede guardar números positivos y el cero, además, sólo permite una única operación que es la suma. Sin embargo, los constructores de ADDITION saben que la suma es suficiente para implementar cualquier operación aritmética básica, al menos con números enteros. Entonces, su labor será la construcción de un programa Python que lea dos números enteros no negativos y que calcule la diferencia entre ellos.

Entradas: Dos números enteros no negativos M y N , es decir, $M \geq 0$ y $N \geq 0$. Si cualquiera de los valores ingresados, M y/o N , son negativos el programa debe solicitar nuevamente su valor hasta que sean ingresados correctamente.

Salida : La resta $M-N$.

Ejemplo de entradas: $M = 24$ y $N = 54$

Salida : $24 - 54 = -30$

- 2.- Un número perfecto es aquél que es igual a la suma de sus divisores, exceptuando él mismo (estos divisores que no incluyen al mismo número son los que se conocen como factores o divisores propios). Simplemente para recordar o aclarar, los divisores de un número natural son los números naturales que lo pueden dividir, resultando de cociente otro número natural y de resto 0, es decir, la división es exacta. Escriba un programa Python que lea un número entero y determine si se trata o no de un número perfecto.

Entradas : La única entrada del algoritmo es un número entero positivo.

Salidas : La única salida del algoritmo es el mensaje `<n> es un número perfecto` o `<n> no es un número perfecto`, donde n es el número ingresado.

Ejemplo de entrada 1: 12

Ejemplo de salida 1: 12 no es un número perfecto

Observación: Los divisores propios de 12 son: 1, 2, 3, 4 y 6. La suma de estos números es $16 \neq 12$.

Ejemplo de entrada 2: 28

Ejemplo de salida 1: 28 es un número perfecto

Observación: Los divisores propios de 28 son: 1, 2, 4, 7 y 14. La suma de estos números es 28.

-
- 3.- Martín es un prodigioso estudiantes a quien las matemáticas le fascinan, siempre está pensando en operaciones raras con la ilusión de encontrar una relación numérica que se convierta en un nuevo descubrimiento.

Lo que en la actualidad le tiene inquieto es saber si existe alguna relación entre la cantidad de ceros y de otros dígitos presentes en el factorial de un número. Hasta ahora ha realizado experimentos con los números más pequeños, esto es:

```
0! es 1, uno tiene 0 ceros y un dígito distinto de cero.
1! es 1, uno tiene 0 ceros y un dígito distinto de cero.
2! es 2, dos tiene 0 ceros y un dígito distinto de cero.
3! es 6, seis tiene 0 ceros y un dígito distinto de cero.
4! es 24, veinticuatro tiene 0 ceros y dos dígitos distintos de cero.
etc.
```

A Martín le interesa tener un cuadro más completo de resultados de esta operación para establecer, si es que la hay, alguna relación.

Escriba un programa en Python que, dado un número entero calcule la cantidad de ceros y de dígitos distintos de cero de su factorial.

Entrada: La única entrada al programa es un número entero mayor o igual a cero.

Salidas: El programa tiene dos salidas, la cantidad de ceros y la cantidad de dígitos distintos de cero del factorial del número ingresado.

Ejemplo entrada 1: 4

Ejemplo salida 1: 0 y 2

Ejemplo entrada 2: 12

Ejemplo salida 2: 4 y 5

Observación: $12! = 479001600$ y este número tiene 4 ceros y 5 dígitos distintos de cero.

- 4.- Implemente un programa Python que calcule la operación potencia (x^y) usando sólo sumas. Los valores ingresados serán x e y , ambos números enteros.

Entradas: $x, x > 0$ e $y, y \geq 0$.

Salidas: La salida está compuesta por un único valor entero correspondiente a x^y .

Ejemplo de entrada 1: 6, 2.

Ejemplo de salida 1: 36.

Ejemplo de entrada 2: 3, 0.

Ejemplo de salida 2: 1.

Ejemplo de entrada 3: 5, 4.

Ejemplo de salida 3: 625.

- 5.- Para realizar una operación de multiplicación, los egipcios empleaban el siguiente método, para cuya explicación supondremos que se desea multiplicar 28×8 .

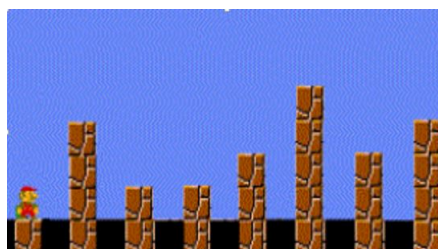
Se anota en 2 columnas los operandos	28	8
luego se registra la mitad de 28 y el doble de 8	14	16
luego se registra la mitad de 14 y el doble de 16	7	32
luego se registra la mitad de 7 (sin fracción) y el doble de 32	3	64
luego se registra la mitad de 3 y el doble de 64	1	128

Luego, se eliminan los números de la segunda columna que están frente a números pares de la primera columna y se suman los números de la segunda columna que no han sido eliminados obteniendo el resultado de la multiplicación:

Se elimina el 8 por estar delante de un 28 (par)	28	-
Se elimina el 16 por estar delante de un 14 (par)	14	-
se mantiene el 32	7	32
se mantiene el 64	3	64
se mantiene el 128	1	128
Resultado		224

Siguiendo este procedimiento, construya un programa Python que efectúe la multiplicación de 2 enteros usando el método de multiplicación Egipcia.

- 6.- Mario (Bros) ha llegado al castillo. Sólo le resta cruzar un conjunto de N murallas para ingresar a la recámara, vencer al monstruo y salvar a la princesa. En este problema estamos interesados en los saltos sobre las murallas. Para ello te indicaremos la altura de las N murallas recorridas desde la izquierda a la derecha. Mario se encuentra parado sobre la primera muralla y debe saltar a la siguiente muralla y así sucesivamente hasta llegar al otro extremo. Esto quiere decir que Mario dará $(N - 1)$ saltos. Un salto hacia arriba ocurrirá cuando Mario avance hacia una muralla más alta y, análogamente, un salto hacia abajo ocurrirá cuando Mario avance hacia una muralla más baja. El problema consiste en realizar un programa Python que determine cuántos saltos hacia arriba y cuántos saltos hacia abajo realizará Mario para ingresar al castillo.



Entradas : La primera línea de entrada contiene un entero N ($0 < N \leq 50$) que corresponde a la cantidad de murallas en el camino de Mario. La siguiente línea de entrada indicará la altura de cada una de las N murallas, desde la de la izquierda a la de la derecha. Cada altura es un entero positivo menor o igual a 10.

Salidas : El programa debe entregar dos números enteros, el primero corresponde a la cantidad de saltos hacia arriba y el segundo corresponde a los saltos hacia abajo.

Ejemplos de entradas 1: 8 1 4 2 2 3 5 3 4

Ejemplos de salida 1:

Saltos arriba : 4
Saltos abajo : 2

Ejemplos de entradas 2: 5 1 2 3 4 5

Ejemplos de salida 2:

Saltos arriba : 4
Saltos abajo : 0