



503202/503203 Programación Programación Básica en Python 5

EQUIPO PROGRAMACIÓN

11 de abril de 2025

Ejercicio de referencia

- A. Para apoyar en matemáticas a su hermanita pequeña que está en la básica, Ud. ha decidido crear un programa Python que le muestre las tablas de multiplicar entre el 1 y la N ($1 \leq N \leq 12$). Cada tabla debe mostrar los resultados de multiplicar el número correspondiente por los números del 1 al 10.

Entrada: La única entrada a este programa es el valor de N , el cual, si se ingresa fuera de rango debe obligar al programa a reingresar hasta que el valor esté correcto.

Salida: La salida a este programa por N tablas de multiplicar entre la 1 y la N , cada una separada por un espacio entre ellas.

Ejemplo de entrada 1:

3

Ejemplo de salida 1:

```
1x1=1
1x2=2
1x3=3
...
3x8=24
3x9=27
3x10=30
```

Solución:

```
N=int(input('N='))
while N<1 or N>12:
    N=int(input('N='))
for i in range(1,N+1):
    for j in range(1,11):
        print(i,"x",j,"=",i*j)
    print()
```

Ahora te toca a tí...

- 1.- Para su tarea de matemáticas, Euclides necesita que, dado dos puntos del espacio bidimensional $a = (a_x, a_y)$ y $b = (b_x, b_y)$, se calcule el punto medio según la fórmula:

$$\frac{a+b}{2} = \left(\frac{a_x+b_x}{2}, \frac{a_y+b_y}{2} \right);$$

e identificar a qué cuadrante del plano pertenece dicho punto medio. Considere que un punto $p = (p_x, p_y)$ se encuentra en el primer cuadrante si $p_x > 0$ y $p_y > 0$, el punto p se encuentra en el segundo cuadrante si $p_x < 0$ y $p_y > 0$, el punto p se encuentra en el tercer cuadrante si $p_x < 0$ y $p_y < 0$, el punto p se encuentra en el cuarto cuadrante si $p_x > 0$ y $p_y < 0$, en otro caso, el punto p se encuentra en un eje.

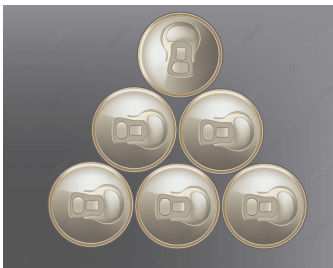
Entrada: La entrada al programa consiste de 4 números enteros sin restricción de valor, correspondientes a las coordenadas a_x , a_y , b_x y b_y .

Salida: La salida del programa es uno de los siguientes mensajes El punto medio es (p_x, p_y) - C cuadrante o El punto medio es (p_x, p_y) - Eje

Ejemplo de Entrada: 5 -5 9 -6

Ejemplo de Salida: El punto medio es (7.0, -5.5) - 2do cuadrante

- 2.- En el supermercado, el reponedor coloca las latas de conserva apiladas triangularmente, esto es, en último piso una lata, en el penúltimo piso dos latas, en el antepenúltimo piso tres, y así sucesivamente. Por ejemplo, seis latas se apilan así:



Sin embargo, no todo número de latas se puede apilar triangularmente. Realice un programa en Python, en el que dada una cantidad de latas n , introducido por el usuario, compruebe si se pueden apilar.

Entrada: La única entrada al programa contiene un entero n ($n \geq 1$) que representa la cantidad de latas.

Se debe validar la entrada, esto significa que si su valor está fuera de rango se debe repetir sucesivamente hasta que el ingreso sea correcto.

Salida: La única salida puede ser uno de los siguientes mensajes Las n latas se pueden apilar en q filas o No apilables. Las n latas se pueden apilar en q filas pero sobran s latas.

Ejemplo de Entrada 1: 6

Ejemplo de Salida 1: Las 6 latas se pueden apilar en 3 filas

Ejemplo de Entrada 2: 8

Ejemplo de Salida 2: No apilables. Las 8 latas se pueden apilar en 3 filas pero sobran 2 latas

-
- 3.- Organilada® es una fábrica de mermeladas orgánicas que, en vez de cosechar del árbol, fabrica mermeladas solo con las frutas que se caen por sí solas, procesándolas apenas tocan el suelo. El problema que tiene Organilada® es que no puede almacenar frascos de mermelada vacíos y desconoce la cantidad que podrá llenar en un día, por tanto, no puede mandarlos a pedir con anticipación. Para resolver este problema se le pide escribir un programa en Python que reciba información sobre las frutas recolectadas y calcule la cantidad de frascos necesarios para embotellar toda la mermelada.

Entrada: Las entradas del programa consiste en varias líneas. La primera contiene T , un entero positivo menor o igual a 1000, que representa la capacidad (en gramos) de los frascos disponibles. La segunda línea contiene M , un entero no negativo menor o igual a 200, el cual representa la cantidad de mermelada (en gramos) que se obtiene por cada fruta. Luego, el programa debe aceptar el ingreso de un conjunto de valores enteros positivos, que representan la cantidad de frutas que caen por día. Las entradas terminan cuando se encuentra un -1.

Si cualquiera de las entradas está fuera de rango, el programa debe desplegar el mensaje “Error, valor fuera de rango”, y dar la posibilidad de reingresar el valor.

Salida: El programa debe desplegar la cantidad máxima diaria de frascos que se pueden llenar de mermelada, considerando que puede haber quedado mermelada de los días anteriores. Además, al terminar el ingreso de frutas, el programa debe imprimir la cantidad restante de mermelada sin embotellar.

Ejemplo de entrada:

```
750
100
5
7
10
2
-1
```

Ejemplo de salida:

```
0
1
1
1
150
```

- 4.- Varios hospitales presentan déficit de médicos en sus urgencias, por lo que se ha planteado el desarrollo de un software que asigne prioridad a sus pacientes. Como parte del equipo de desarrollo tecnológico del MINSAL se les pide que diseñen un programa en Python para la asignación de prioridad basado en las respuestas dadas por el paciente a un cuestionario. Todas las preguntas del cuestionario se responden sólo con un si o con un no y, dependiendo de esta respuesta, se asigna un valor correspondiente a la prioridad. El cuestionario contiene las siguientes preguntas:

- a) ¿Sospecha estar enfermo de COVID-19?
- b) ¿Sufrió algún accidente de gravedad y presenta quemadura, fractura o corte de tejido?
- c) ¿Siente dificultad al respirar?
- d) ¿Es asmático?
- e) ¿Siente dolor al pecho?
- f) ¿Los síntomas que lo traen a la urgencia llevan más de un día en igual o mayor magnitud?

Las prioridades se asignan según las siguientes reglas:

Si la respuesta a la pregunta (a) es positiva, la prioridad del paciente es 2, sin importar las otras respuestas.

Si la respuesta a la pregunta (b) o (e) es positiva, la prioridad del paciente es 1.

Si la respuesta a la pregunta (c) y (d) son positivas, la prioridad del paciente es 1.

Si la respuesta a la pregunta (c) es positiva y (d) es negativa, la prioridad del paciente es 3.

Si las respuestas a las preguntas (a), (b), (c) y (e) son negativas y la respuesta a la pregunta (f) es positiva, la prioridad del paciente es 4.

Entradas: Existen 6 líneas de entrada, las cuales corresponden a las respuestas a las preguntas a, b, c, d, e y f. Las respuestas a estas preguntas puede ser sí o no.

Salidas: La salida del programa corresponde al valor de la prioridad asignado, el cual puede ser 1 (prioridad máxima), 2 (derivación a equipo COVID), 3 (prioridad intermedia) o 4 (prioridad baja).

Ejemplo de entrada 1: Sí No No Sí No Sí

Ejemplo de salida 1: Prioridad 2

Ejemplo de entrada 2: No No Sí No No No

Ejemplo de salida 2: Prioridad 3.

Ejemplo de entrada 3: No Sí No No No No

Ejemplo de salida 3: Prioridad 1.

- 5.- Fabián está de cumpleaños y este año cumple n años, por lo tanto, tendrá n velas en su torta. Como es el hermano menor y como todos los años, las velas son recicladas, por lo que tienen diferentes alturas. Así, cuando Fabián sopla sus velas, las más grandes no dejan que el aire llegue al resto de las velas, así que sólo las velas más altas terminan apagándose con su soplo.

Entonces, dada la altura de n velas, genere un programa Python, que permita encontrar la cantidad total de velas que efectivamente se pueden apagar con un soplo de Fabián.

Entrada: La primera línea contiene un entero n (donde $1 < n \leq 100$), que representa la cantidad de velas que tendrá la torta. Las siguientes n líneas contienen, cada una, un entero que representa la altura h de una vela ($0 < h \leq 15$).

Se debe validar las entradas, esto significa que si su valor está fuera de rango se debe repetir sucesivamente hasta que el ingreso sea correcto.

Salida: La cantidad total de velas que se apagan.

Ejemplo de Entrada 1: 4 – 3,2,1,3 por simplicidad los valores se han puesto en una línea

Ejemplo de Salida 1: 2

Ejemplo de Entrada 2: 8 – 10,12,8,12,2,4,10,12

Ejemplo de Salida 2: 3

- 6.- Daniela quiere configurar un panel que contenga diferentes números con LEDs (Diodos Emisores de Luz, por sus siglas en inglés). Lamentablemente Daniela no posee muchos LEDs y no está seguro de si será capaz de montar un número deseado. Entonces, considerando las configuraciones de LEDs para cada uno de los dígitos mostrados

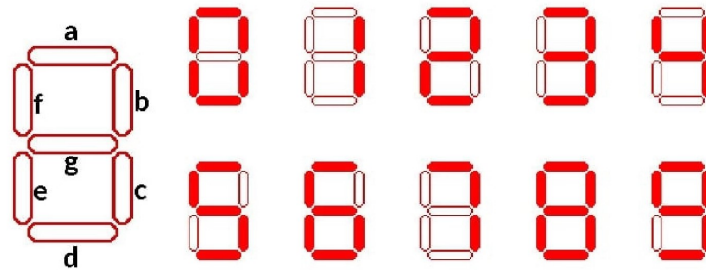


Figura 1: LEDs necesarios para cada dígito.

en la figura, realice un algoritmo usando diagramas de flujo que calcule la cantidad de LEDs requeridos para montar un número ingresado.

Entrada: La entrada contiene un entero N ($1 \leq N \leq 2000$), correspondiente al número que Daniela quiere representar con LEDs.

Salida: Despliega una línea que contenga el número de LEDs que Daniela necesita para establecer el valor deseado, seguido por la palabra "LEDs".

Ejemplo entrada 1: 100

Ejemplo salida 1: 14 LEDs

Ejemplo entrada 2: 1234

Ejemplo salida 2: 16 LEDs

- 7.- Méndez y Soto (MENTO Asociados) son inversionistas interesados en invertir en una empresa de energías renovables que cotiza en la bolsa de valores. MENTO Asociados ha conseguido los valores de las acciones de la empresa en los últimos N días y se dispone a analizarlos. Para ello le contratan con el fin de que diseñe un algoritmo usando diagramas de flujo, el cual debe leer los valores de las acciones y reportar los valores peak y su valor promedio.

Los valores peak se definen como los puntos más altos de una serie de datos, cuyo antecesor y sucesor son de menor magnitud. También se considera peak a un valor cuyo antecesor es de menor magnitud y su(s) sucesor(es) son de igual magnitud, todos seguidos por un número de menor magnitud.

Entradas: En la primera línea viene un entero N ($N \geq 3$). Luego, en las líneas de la 2 a la $N + 1$, se ingresan los N valores de las acciones de la empresa de interés (estos datos deben ser ingresados individualmente).

Salidas: La salida del programa está compuesta por el conjunto de valores peak y el valor promedio de las acciones.

Ejemplo de entrada 1: 10 5.41 6.87 9.25 4.75 4.31 4.57 7.42 8.74 8.74 6.65

Ejemplo de salida 1: Peaks: 9.25, 8.74 Promedio: 6.671

Ejemplo de entrada 2: 6 5.41 3.02 8.45 8.45 5.03 9.17

Ejemplo de salida 2: Peaks: 8.45 Promedio: 6.59