

Plan Formativo Full Stack Python	
Módulo	Programación Básica en Python
Tema	Confección de un abaco digital
Nivel de Dificultad	Alta
Ejecución	Grupal (3 - 4 personas)
Intención del aprendizaje o aprendizaje esperado:	
<ul style="list-style-type: none"> Manejo de variables de tipo número, string, lista, diccionario. Estructuración del proyecto en funciones de cálculo, despliegue, almacenamiento de datos, interrogación a usuario, etc. Lógica matemática para descomposición de números en unidades, decenas, centenas, etc. Trabajo en grupo de desarrollos convergentes. 	
Planteamiento del Problema:	
<p>Haga un programa que muestre una cantidad ingresada por el usuario en forma de abaco en pantalla:</p> <p>En el ejemplo de abajo un usuario ingresó la cantidad 841.231, y se debería desplegar la siguiente imagen en pantalla</p>	
<p>El diagrama muestra un abaco digital con seis columnas. Cada columna tiene diez varillas (representadas por líneas verticales). Las varillas se activan con 'X' para indicar la cantidad. Debajo de cada columna se muestra el valor numérico: 100.000, 10.000, 1.000, 100, 10 y 1.</p>	
<p>Utilice un diccionario que almacene las unidades, decenas, centenas, unidades de mil, unidades de diez mil y unidades de cien mil.</p>	

Programación Básica en Python

Confección de un abaco digital

Alta

Grupal (3 - 4 personas)

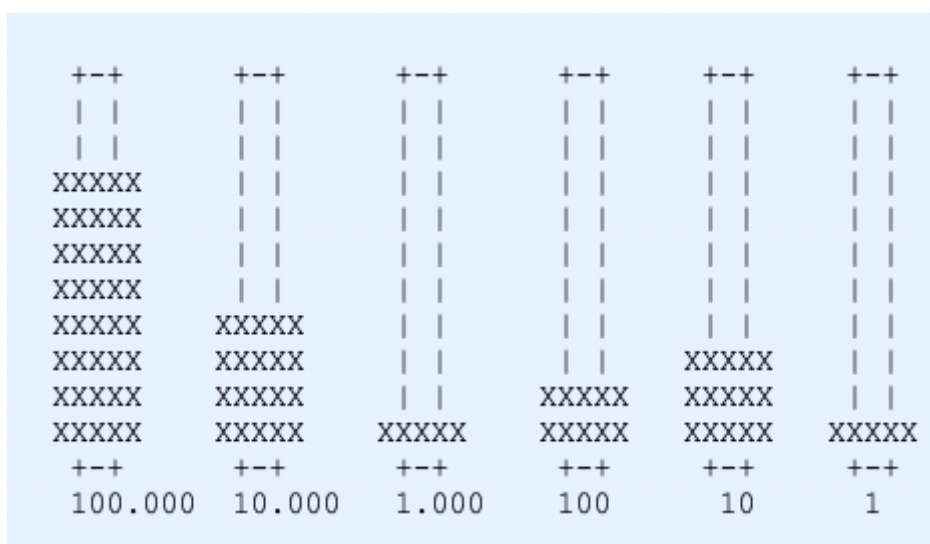
Intención del aprendizaje o aprendizaje esperado:

- Manejo de variables de tipo número, string, lista, diccionario.
- Estructuración del proyecto en funciones de cálculo, despliegue, almacenamiento de datos, interrogación a usuario, etc.
- Lógica matemática para descomposición de números en unidades, decenas, centenas, etc.
- Trabajo en grupo de desarrollos convergentes.

Planteamiento del Problema:

Haga un programa que muestre una cantidad ingresada por el usuario en forma de abaco en pantalla:

En el ejemplo de abajo un usuario ingresó la cantidad 841.231, y se debería desplegar la siguiente imagen en pantalla



Utilice un diccionario que almacene las unidades, decenas, centenas, unidades de mil, unidades de diez mil y unidades de cien mil.

Cada intento debe ser almacenado en una lista, tal lista debe contener todos los intentos de usuario.

Si el usuario desea dejar de ejecutar el programa en cualquier instante, permítale terminar escribiendo 'salir'.

Previo a 'salir' muestre en pantalla todos los intentos de la lista, por ejemplo:

```
841.231
10.231
123
903.267
etc.
```

No olvide ubicar el punto separador de miles para mostrar la cifra más claramente.

Haga su código legible y fácilmente ejecutable, dando las instrucciones de su uso.

Separe las distintas partes en funciones propiamente nombradas y de una extensión razonable.

Escriba todos los comentarios que estime convenientes.

No solo se evaluará que el programa haga lo requerido, también se evaluará el orden del código. Códigos poco legibles no se evaluarán positivamente.

Recursos Bibliográficos :

[1] Modulo 2: Programación Básica en Python