**Codificación y Programación. Juan David González**

**Quiz Capítulos #3 Y #4.**

**Q. 03-01.** Diseñe un programa que reciba las coordenadas (x1, y1), (x2, y2) de dos puntos del usuario e imprima la distancia entre los dos puntos. Para hacer esto, implemente la función distancia (x1, y1, x2, y2). Pauta de código: consulte la ecuación para encontrar la distancia entre dos puntos.

import math

print("ingrese los valores para la coordenada 1")

x1 = int(input("ingrese el valor de x1:"))

y1 = int(input("ingrese el valor de y1:"))

print("ingrese los valores para la coordenada 2")

x2 = int(input("ingrese el valor de x2:"))

y2 = int(input("ingrese el valor de y2:"))

distancia = math.sqrt((x2-x1)\*\*2+(y2-y1)\*\*2)

print("la distancia entre la coordenada 1 y la coordenada 2 es =", distancia)

**Q. 03-02.** Un palíndromo es una oración, palabra o cadena que se lee igual al derecho o al revés. Por ejemplo, reconocer, radar o la frase “anita lava la tina”. Usemos una llamada recursiva para determinar el palíndromo. Defina una función llamada is\_palindrome y escriba un programa que reciba una cadena del usuario e imprima si el palíndromo es correcto o no. Pauta de código: Llame a la función is\_palindrome dentro de la función is\_palindrome(función recursiva).

word = input('Ingrese la cadena')

if str(word) == str(word)[::-1] :

    print("Palindrome")

else:

    print("No es Palindrome")

**Q. 03-03.** Defina una lista llamada n\_list con valores de [10, 20, 30]. Utilice la función map y lambda para imprimir los resultados de duplicar, triplicar y cuadruplicar los objetos de la lista

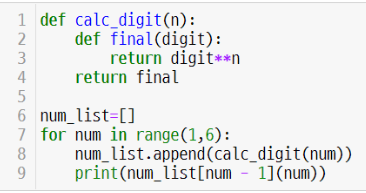
ejemplo:mapped\_numbers = list(map(lambda x: x \* 2 + 3, numbers))

n\_lista = [10,20,30]

mapped\_numbers = list(map(lambda x: x \* 2 + 3, n\_lista))

print(mapped\_numbers)

**Q. 03-04.** Explique el funcionamiento del siguiente código, para ello aplique una prueba de escritorio hecha “a mano”



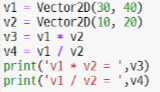
**Pues mas que todo, se define una lista llamada “num\_lista”, que se encuentra vacía. El ciclo for que tiene, le permite imprimir un valor en cada iteración que se da en un rango de 5 posiciones. Agregándole sucesivamente la multiplicación del numero de interación, por el valor que se encuentra en N posición de la lista.**

**Q. 03-05.** Implemente las funciones de multiplicación (\*) y división (/) de dos vectores usando los métodos especiales \_\_mul\_\_ y \_\_truediv\_\_. Suponiendo que v1 es (30, 40) y v2 es (10, 20), codifique para devolver el siguiente resultado como resultado de la multiplicación y división de dos vectores.



Pauta de codificación: escriba un código implementando clases que reciba 2 objetos de la clase vector2D y los opere como se ha solicitado

Hint:



import operator

v1 = [30,40]

v2 = [10,20]

for x in v1:

  for y in v2:

    print('La multiplicación de ', x ,' y ' , y,' es: ' + str(operator.mul(x, y)))

    print('La división de ', x ,' y ' , y,' es: ' + str(operator.truediv(x, y)))

**Q. 04-01.** La siguiente es la implementación de una pila en python. ¿Cuál será el resultado del siguiente código?

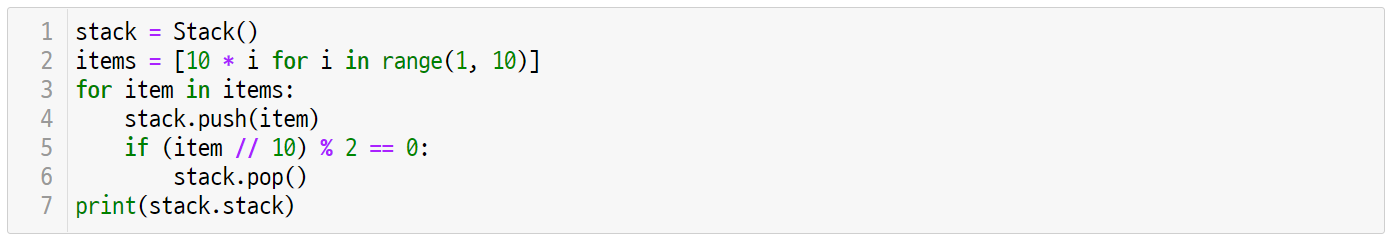


Pauta de codificación: escriba cual es el resultado esperado de ejecutar cada línea (prueba de escritorio)

**Lo que se espera en cada ejercicio, es agregar banana, apple, tomato, strawberry y grapes. Pero en Python no se cuenta con la función push para agregar, se podría utilizar append en su lugar.**

**En todo caso si esto se ejecutara de manera correcta el resultado sería: se agregaría banana, apple, tomato, luego con el pop, eliminamos tomato. Posteriormente se agrega strawberry y grapes, y el pop elimina grapes.**

**Q. 04-02.** La siguiente es la implementación de una pila en python. ¿Cuál será el resultado del siguiente código?

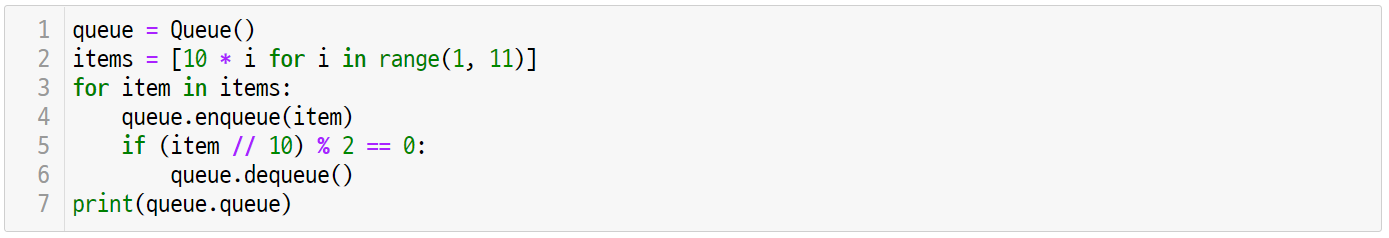


Pauta de codificación: escriba cual es el resultado esperado de ejecutar cada línea (prueba de escritorio)

**En estas líneas de código se trata de introducir los datos mediante un ciclo for del mismo diccionario.**

**Ya luego mediante un ciclo for se hace la función de agregar los valores del diccionario creado en otra variable que los almacene, para luego condicionarla por un if que si es igual a 0 el modulo de la división , se elimina ese valor y procede a imprimirse completo el diccionario.**

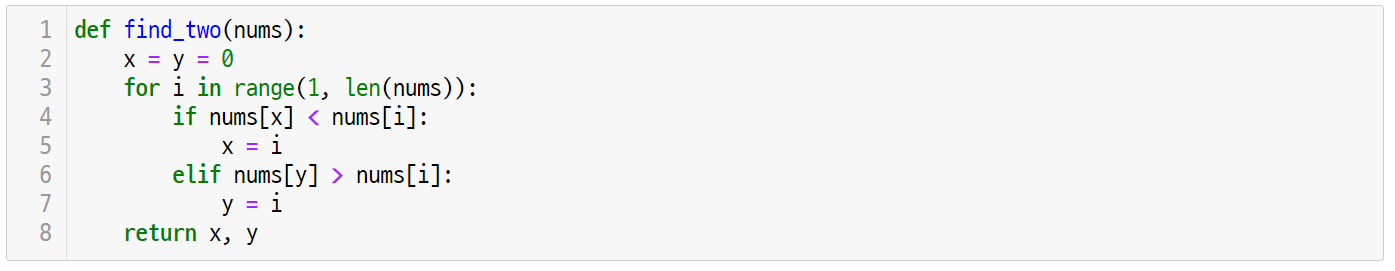
**Q. 04-03.** A continuación se muestra la implementación de una cola en python(queue). ¿Cuál será el resultado del siguiente código?



Pauta de codificación: escriba cual es el resultado esperado de ejecutar cada línea (prueba de escritorio)

**La función queue nos permite agregar valores a una cola que esté llena, por lo que en el ciclo for se agrega los valores del diccionario ítems. Luego en la condición se hacer que la división del modulo es igual a 0, se procede a eliminar ese valor de la cola con la función dequeue y procede a imprimirse la cola final**

**Q. 04-04.** ¿Cuál es el algoritmo de la siguiente función find\_two()? Analice el código y escriba el resultado de la ejecución.



Texto

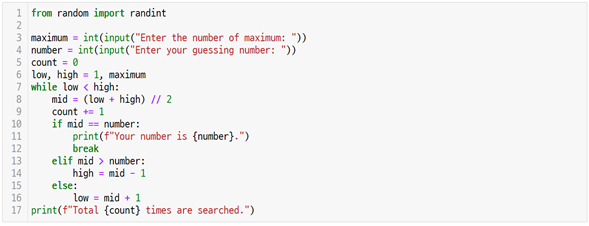
Descripción generada automáticamente

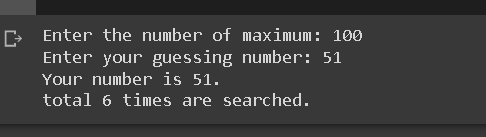
**Aquí la función find\_two, hace el proceso de ver que se cumpla la condición de ver que numero en x e y es mayor y menor que el otro, y si se cumple la condición, se establece el valor en x e y, para luego retornar eso valores e imprimirlos en el print final**

**Q. 04-05.** ¿Cuántas comparaciones debe realizar la función find\_two() implementada en la pregunta anterior (Q.04-04)?

**En ciclo for hace 8 comparaciones de uno en uno para luego imprimir los valores de x e y**

**Q. 04-06.** El siguiente es el código para el juego de combinación de números. Si el máximo es 100 y el número es 51, ¿cuál es la salida de count?





**Q. 04-07.** En el código del juego de combinación de números, si el máximo es 100 y el número es 25, ¿cuál es el resultado del conteo?

Texto

Descripción generada automáticamente

**Q. 04-08.** Usando la función insert de la clase hash\_table, ingrese la clave "Alicia en el país de las maravillas", a continuación obtenga la clave hash de buscar dicha clave dentro de la hash\_table

Pauta de codificación: Ayúdese de la definición de la clase hash\_table del siguiente ejemplo: <https://pythondiario.com/2018/06/tabla-hash-en-python.html>

class hash\_table:

    def \_\_init\_\_(self):

        self.table = [None] \* 127

    # Función hash

    def Hash\_func(self, value):

        key = 0

        for i in range(0,len(value)):

            key += ord(value[i])

        return key % 127

    def Insert(self, value): # Metodo para ingresar elementos

        hash = self.Hash\_func(value)

        if self.table[hash] is None:

            self.table[hash] = value

    def Search(self,value): # Metodo para buscar elementos

        hash = self.Hash\_func(value);

        if self.table[hash] is None:

            return None

        else:

            return hex(id(self.table[hash]))

    def Remove(self,value): # Metodo para eleminar elementos

        hash = self.Hash\_func(value);

        if self.table[hash] is None:

            print("No hay elementos con ese valor", value)

        else:

            print("Elemento con valor", value, "eliminado")

            self.table[hash] is None;

H = hash\_table()

H.Insert("Alicia")

H.Insert("en")

H.Insert("el")

H.Insert("país")

H.Insert("de")

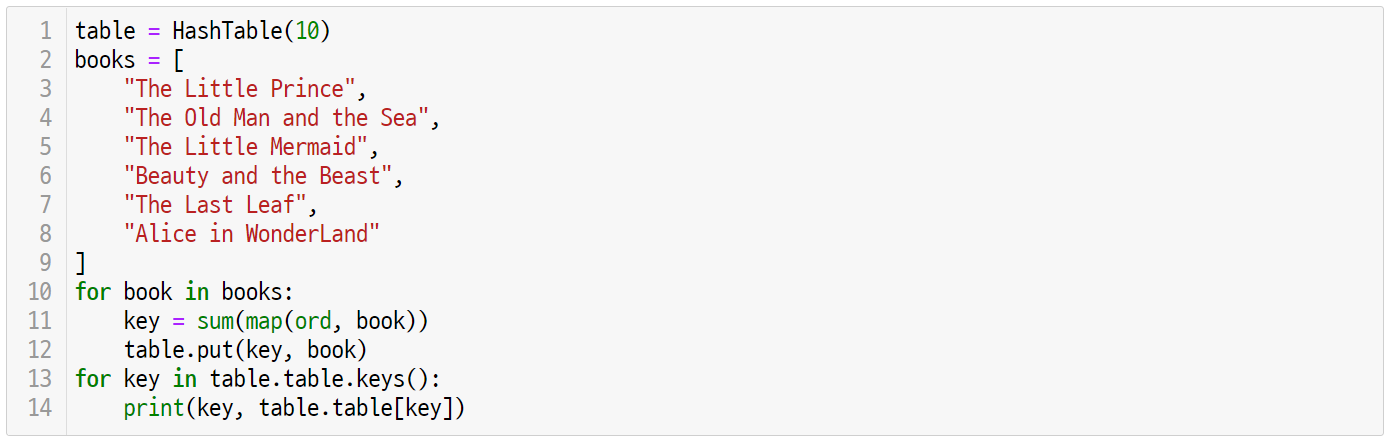
H.Insert("las")

H.Insert("maravillas")

print(H.Search("Alicia"))

print(H.Remove("las"))

**Q. 04-9.** Si la nueva estantería tiene 10 compartimentos, usa el siguiente código para averiguar qué libro hay en cada compartimento.



**En este código se me dificultó ejecutarlo en el editor de código, porque tiene errores que no entiendo.**

**Lo que si pude notar es que esta tupla tiene 10 espacios, pero solo tiene 6 estanterías con datos, por lo que algunos no tendrán libros.**