



Fund Programacion G16-17-18-19-20-21

Descripción

Editar

Ver entrega

Reto 2 - Semana 4 - Variante 3

Ficheros requeridos: variante3.py ([Descargar](#))

Tipo de trabajo: Individual

[CLICK AQUÍ PARA LEER EL ENUNCIADO](#)

Ficheros requeridos

variante3.py

```
1 import numpy as np
2 #NOTA: PARA ESTE RETO PUEDES PROBAR TU PROGRAMA, DANDO CLICK EN LA NAVE ESPACIAL
3 #LA CONSOLA TE DARÁ INSTRUCCIONES SI PUEDES EVALUAR O NO TU CÓDIGO
4 |
5 #En este espacio podrás programar las funciones que deseas usar en la función solución (NO ES OBLIGATORIO CREAR OTRAS FUNCIONES)
6 """Inicio espacio para programar funciones propias"""
7
8
9
10 #PUEDES PROGRAMAR CUANTAS FUNCIONES DESEES
11
12
13
14 """Fin espacio para programar funciones propias"""
15
16 def solucion(A):
17     """
18     En esta función deberás programar tu solución.
19     Explicación del parámetro que recibe:
20     - A: Es una matriz cuadrada de números enteros aleatoria (NO TE DEBES PREOCUPAR POR LLENARLA, YA LO ESTÁ),
21       para indexar un elemento en la fila i, columna j se hace así:
22       A[i,j] o A[i][j], como te sientas más cómodo.
23       El orden de la matriz lo puedes calcular así: n = A.shape[0]
24
25     Explicación de lo que debe retornar:
26     -sum_diagonal_principal: Suma de los elementos de la diagonal principal, es de tipo float o entero (Elige el tipo que quieras).
27     -prod_diagonal_secundaria: Producto de los elementos de la diagonal secundaria, es de tipo float o entero (Elige el tipo que quieras)
28     -resultado2_mod_resultado1: Módulo entre prod_diagonal_secundaria y sum_diagonal_principal, es de tipo float o entero (Elige el tipo que quieras)
29     """
30
31
32
33
34
35
36
37     return sum_diagonal_principal, prod_diagonal_secundaria, resultado2_mod_resultado1
38
39
40
41 """
42 Estas líneas de código que siguen, permiten probar si su ejercicio es correcto
43 """
44 #NO MODIFICAR
45 matriz_entrada = np.array([[73, 13, 6, 1, 29],
46                             [28, 72, 76, 86, 48],
47                             [94, 18, 32, 24, 33],
48                             [63, 11, 16, 69, 40],
49                             [38, 20, 45, 78, 61]])
50 dp_correcto = 307
51 ds_correcto = 33359744
52 ds_mod_dp_correcto = 203
53
54 dp_estudiante = solucion(matriz_entrada)[0]
55 ds_estudiante = solucion(matriz_entrada)[1]
56 ds_mod_dp_estudiante = solucion(matriz_entrada)[2]
57
58 print("MATRIZ ENTREGADA:\n", matriz_entrada, "\n")
59 print(f"""====SALIDA ESPERADA====
60 Suma de los elementos de la diagonal principal = {dp_correcto}
61 Producto de los elementos de la diagonal secundaria = {ds_correcto}
62 {ds_correcto} mod {dp_correcto} = {ds_mod_dp_correcto}""")
63
64 print(f"""====TU SALIDA====
65 Suma de los elementos de la diagonal principal = {dp_estudiante}
66 Producto de los elementos de la diagonal secundaria = {ds_estudiante}
67 {ds_estudiante} mod {dp_estudiante} = {ds_mod_dp_estudiante}\n""")
68
69 if dp_correcto == dp_estudiante and ds_correcto == ds_estudiante and ds_mod_dp_correcto == ds_mod_dp_estudiante:
70     print("Todo se ve correcto, ¡Procede a calificar tu código!")
71 else:
72     print("Las salidas no coinciden, ¡Estás olvidando algo en tu código!")
73
```

[Servicio Web](#)

VPL



Datos de contacto

✉ soportemisiontic@udea.edu.co
🌐 ingeniaudea.edu.co/portal

Síguenos en nuestras redes



La información aquí recolectada, será tratada conforme a los estatutos de propiedad intelectual de la Universidad de Antioquia, y en consideración a la Política de tratamientos de datos, la cual puede consultar en: <https://goo.gl/mqBe7p>

[Descargar la app para dispositivos móviles](#)