Лабораторна робота №7

Варіант 6

**Завдання 1**

Визначити суму додатних елементів матриці з непарною сумою індексів.

n = int(input("n = "))  
m = int(input("m = "))  
  
matrix = [  
 [int(input("mat[{0}][{1}] = ".format(i,j))) for j in range(m)] for i in range(n)  
]  
  
product = 0  
  
  
for i in range(0, n):  
 for j in range(0, m):  
  
 if matrix[i][j] > 0 and (i + j)% 2 == 1:  
 product += matrix[i][j]  
  
print(product)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| n = 2  m = 2  mat[0][0] = 1  mat[0][1] = 2  mat[1][0] = 3  mat[1][1] = 4 | 5 |
| n = 2  m = 3  mat[0][0] = 8  mat[0][1] = 7  mat[0][2] = 6  mat[1][0] = 2  mat[1][1] = 9  mat[1][2] = 7 | 16 |

**Завдання 2**

# Побудувати прямокутно матрицю A еленти якої задаються формулою:aij = i(sin(ix) + cos(jx)), i=1,N, j = 1,M  
# Обчислити добуток елементів, добуток індексів яких менший за задане число х  
  
  
from math import sin, cos  
n = int(input("Кількість стовпців = "))  
m = int(input("Кількість рядків = "))  
x = int(input("x = "))  
  
  
def a\_i\_j(i, j, x):  
 a = i \* (sin(i \* x) + cos(j \* x))  
 return a  
  
  
A = [[a\_i\_j(i, j, x)for i in range(1, n + 1)]for j in range(1, m + 1)]  
for i in A:  
 print(i)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| Кількість стовпців = 2  Кількість рядків = 2  x = 1 | [1.3817732906760363, 2.899199465387643]  [0.4253241482607541, 0.9863011805570786] |
| Кількість стовпців = 5  Кількість рядків = 2  x = 5 | [-0.6752620891999122, -0.5207178508522871, 2.801850076861029, 4.786429744763415, 0.756552176827266]  [-1.797995803739591, -2.766185279931644, -0.5663510667580068, 0.2954948866047009, -4.857116395871127] |

**Завдання 3**

Дано матрицю А і ветор в .Знайти вектор вА  
  
  
n = int(input("Кількість рядків : "))  
m = int(input("Кількість стовпців: "))  
matrix = [[int(input("Element[{0}][{1}]= ".format(i, j))) for j in range(m)] for i in range(n)]  
for i in matrix:  
 print(i)  
vec = [int(input("Елемент вектора: ")) for i in range(m)]  
print("Вектор= " + str(vec))  
answer = []  
s = 0  
matrixt = list(zip(\*matrix)) #Створюємо транспоновану матрицю  
k = -1  
for i in range(m):  
 k += 1  
 if k == m:  
 break  
 for j in range(n):  
 s += vec[k] \* matrixt[i][j]  
 answer.append(s)  
 s = 0  
  
print("Новий вектор = {0}".format(answer))

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| Кількість рядків : 2  Кількість стовпців: 2  Element[0][0]= 3  Element[0][1]= 6  Element[1][0]= 9  Element[1][1]= 8  [3, 6]  [9, 8]  Елемент вектора: 5  Елемент вектора: 9 | Вектор= [5, 9]  Новий вектор = [60, 126] |
| Кількість рядків : 3  Кількість стовпців: 6  Element[0][0]= 1  Element[0][1]= 2  Element[0][2]= 3  Element[0][3]= 4  Element[0][4]= 5  Element[0][5]= 6  Element[1][0]= 9  Element[1][1]= 8  Element[1][2]= 7  Element[1][3]= 5  Element[1][4]= 5  Element[1][5]= 5  Element[2][0]= 5  Element[2][1]= 5  Element[2][2]= 5  Element[2][3]= 5  Element[2][4]= 5  Element[2][5]= 5  [1, 2, 3, 4, 5, 6]  [9, 8, 7, 5, 5, 5]  [5, 5, 5, 5, 5, 5]  Елемент вектора: 5  Елемент вектора: 5  Елемент вектора: 8  Елемент вектора: 2  Елемент вектора: 6  Елемент вектора: 85 | Вектор= [5, 5, 8, 2, 6, 85]  Новий вектор = [75, 75, 120, 28, 90, 1360] |

**Завдання 4**

Розмістити елементи непарних рядків у порядку спадання.  
  
  
A = [  
 [9, 5, 3, 0],  
 [0, -7, 1, 4],  
 [5, 0, 8, 3],  
 [0, -3, 5, 1]  
]  
# Створюємо відсортований список списків  
lis = []  
for el in A[::2]:  
 B = [sorted(el, reverse=True)]  
 lis += B  
 # Замінюємо списки на відсортовані  
A[::2] = lis  
for i in A:  
 print(i)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані (y) |
| [9, 5, 3, 0], [0, -7, 1, 4], [5, 0, 8, 3], [0, -3, 5, 1] | [9, 5, 3, 0]  [0, -7, 1, 4]  [8, 5, 3, 0]  [0, -3, 5, 1] |

Завдання 5

Дана цілочислова прямокутна матриця.Визначити суми елементів в тих стовпцях які містять хочб один відємний елемент  
  
  
n = int(input("Кількість рядків та стовпців: "))  
A = [[int(input("El[{0}][{1}]= ".format(i, j))) for j in range(1, n + 1)] for i in range(1, n + 1)]  
matrix = [[0] \* n for i in range(n)]  
for j in range(n):  
 for i in range(n):  
 matrix[j][i] = A[i][j]  
for i in matrix:  
 print(" " + str(i))  
collumn = 0  
for el in matrix:  
 count = 0  
 m = list(filter(lambda x: x < 0, el))  
 if len(m) < 1:  
 collumn += 1  
 continue  
 else:  
 collumn += 1  
 count = sum(el)  
 print("Сума чисел стовпця № {0}: {1}".format(collumn, count))

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані (y) |
| Кількість рядків та стовпців: 2  El[1][1]= 1  El[1][2]= 2  El[2][1]= 3  El[2][2]= 4 | [1, 3]  [2, 4] |

Завдання 6

Дана числова прямокутна матриця.Визначити номера рядків і стовпців всіх сідлових точок матриці, матриця А має сідлову точку А0 якщо Аij ємвнімальним елементом в і-у рядку і максимальним в j-у стовпці  
def saddle\_point(matrix):  
 if len(matrix) == 1:  
 return 0  
 y = 0  
 while len(matrix) > y:  
 for i in matrix:  
 l\_min = min(i)  
 l\_index = i.index(l\_min)  
 for j in matrix:  
 if l\_min > j[l\_index]:  
 return matrix.index(i), l\_index  
 return None  
 y += 1  
  
  
b = saddle\_point([[0, 850, -1], [1, 2, -2]])  
print(b)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані (y) |
| [1, 850, -1],  [1, 2, -2] | (0, 2) |