

Министерство цифрового развития, связи и массовых коммуникаций
Ордена Трудового Красного Знамени федеральное государственное
бюджетное

образовательное учреждение высшего образования
«Московский технический университет связи и информатики»

Кафедра Математической кибернетики и информационных технологий

Отчёт по задачам на строки
по дисциплине «Сиаод»

Выполнил: студент группы
БВТ1902

Сорокин Никита Андреевич

Москва

2021

Цель работы

1. Массив `A` состоит из целых положительных чисел — длин отрезков. Составьте из трех отрезков такой треугольник, чтобы его периметр был максимально возможным. Если невозможно составить треугольник с положительной площадью — функция возвращает 0.
2. Дан массив неотрицательных целых чисел `nums`. Расположите их в таком порядке, чтобы вместе они образовали максимально возможное число.
3. Дана матрица `mat` размером $m * n$, значения — целочисленные. Напишите функцию, сортирующую каждую диагональ матрицы.

Код программы

```
def perim(a):
    print(a)

    p = 0
    d = 0

    for i in range(len(a)):
        if a[i] > 0:
            d += 1
            for j in range(i + 1, len(a)):
                for z in range(j + 1, len(a)):
                    if a[i] + a[j] > a[z] and a[j] + a[z] > a[i] and a[i] + a[z] > a[j]:
                        k = a[i] + a[j] + a[z]
                        if k > p:
                            p = k

    if p <= 0 or d < 3:
        print("0")
    else:
        print(p)
```

```
def maxchislo(mass):
    print("Исходный массив: ")
    print(mass)
    jmass = []
    imass = []

    for j in range(0, len(mass)):
        for i in range(0, len(mass) - 1):
            jmass.append(mass[i])
            imass.append(mass[i + 1])
            strj = ".join(map(str, jmass))
            stri = ".join(map(str, imass))
            jmass = list(strj)
            imass = list(stri)

        if len(jmass) == 3:
            mass[i], mass[i + 1] = mass[i + 1], mass[i]

    if jmass[0] < imass[0] and len(imass) != 3 and len(jmass) != 3:
```

```

    mass[i], mass[i + 1] = mass[i + 1], mass[i]

    if jmass[0] == imass[0] and ((len(imass) < len(jmass)) or (len(imass) > len(jmass)))
and len(
    imass) != 3 and len(jmass) != 3:
    if len(imass) == 2:
        if imass[0] < imass[1]:
            mass[i], mass[i + 1] = mass[i + 1], mass[i]
        else:
            if jmass[0] > jmass[1]:
                mass[i], mass[i + 1] = mass[i + 1], mass[i]

    if jmass[0] == imass[0] and len(imass) == len(jmass) and len(imass) > 1 and
len(jmass) > 1 and jmass[1] < \
    imass[1] and len(imass) != 3 and len(jmass) != 3:
        mass[i], mass[i + 1] = mass[i + 1], mass[i]

    jmass = []
    imass = []

print("Результат: " + ".join(map(str, mass)))

def sortdig(a):
    k=0

    if len(a) < len(a[1]):
        a = list(map(list, zip(*a)))
        k = 1

    for z in range(0, len(a)-1):
        for i in range(0, len(a)-1):
            for j in range(0, len(a[i]) - 1):
                if a[i][j] < a[i + 1][j + 1]:
                    a[i][j], a[i + 1][j + 1] = a[i + 1][j + 1], a[i][j]

    if k == 1:
        a = list(map(list, zip(*a)))

    print(" ")
    print("Результат: ")

    for i in range(len(a)):
        for i in range(a[j][i]):

```

```
print(a[i])
```

```
print("Задача №1")
```

```
perim([3, 2, -9, -7])
```

```
print("")
```

```
print("Задача №2")
```

```
maxchislo([3,30,34,5,9])
```

```
print("")
```

```
print("Задача №3")
```

```
sortdig( [[11, 25, 66, 1, 69, 7], [23, 55, 17, 45, 15, 52], [75, 31, 36,44, 58, 8], [22, 27, 33, 25,  
68, 4], [84, 28, 14, 11, 5, 50]])
```

Скриншоты работы программы

```
Задача №1  
[3, 2, -9, -7]  
0
```

Рис. 1 – Результат выполнения задания №1

```
Задача №2  
Исходный массив:  
[3, 30, 34, 5, 9]  
Результат: 9534330
```

Рис. 2 – Результат выполнения задания №2

```
Задача №3  
  
Результат:  
[55, 68, 66, 15, 69, 7]  
[33, 36, 50, 58, 8, 52]  
[75, 31, 25, 44, 45, 1]  
[28, 27, 23, 11, 25, 4]  
[84, 22, 14, 11, 5, 17]
```

Рис. 3 – Результат выполнения задания №3

Вывод

В ходе выполнения лабораторной работы, я написал алгоритмы, которые решает задачи на треугольник с максимальным периметром, на составление максимального числа, на сортировки диагоналей в матрице.