Задание 26 (№1) Сибигатулин Рамиль https://vk.com/infmat100

Условие:

На премьеру нового фильма пришли практически все жители небольшого города. Известны номера ряда и места выкупленных билетов. Зал имеет форму квадрата, причём кресла, расположенные на диагоналях квадрата ярко-красного цвета, а остальные – темно-красного. Таким образом, зал диагоналями разбивается на четверти. Оказалось, что в этот день празднуется день «Самой занятой четверти кинозала». Всем, кто на этой четверти полагается по окончании киносеанса вернуть деньги за просмотр (Если таких четвертей будет несколько, то деньги вернут тем людям, которые сидели в четверти с минимальным номером, дабы напомнить о несправедливости этого мира). Определите, день какой четверти будет праздноваться в этом году (верхняя – 1, правая – 2, нижняя – 3, левая - 4) и скольким посетителям вернут деньги, с учетом того, что люди, кресла которых ярко-красные, НЕ могут претендовать на возврат денег?

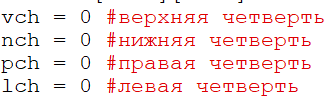
В первой строке входного файла содержится число N – количество рядов в кинозале, во второй строке M – общее количество занятых мест, а в остальных строках по два числа: номер ряда и места выкупленного билета.

В ответ запишите два числа: сначала номер самой занятой четверти зала (цифру 1-4) и количество людей на ней.

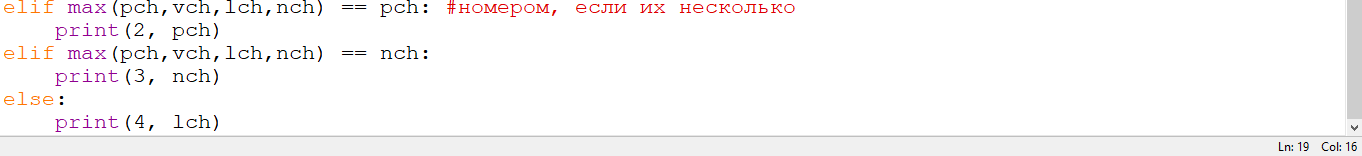
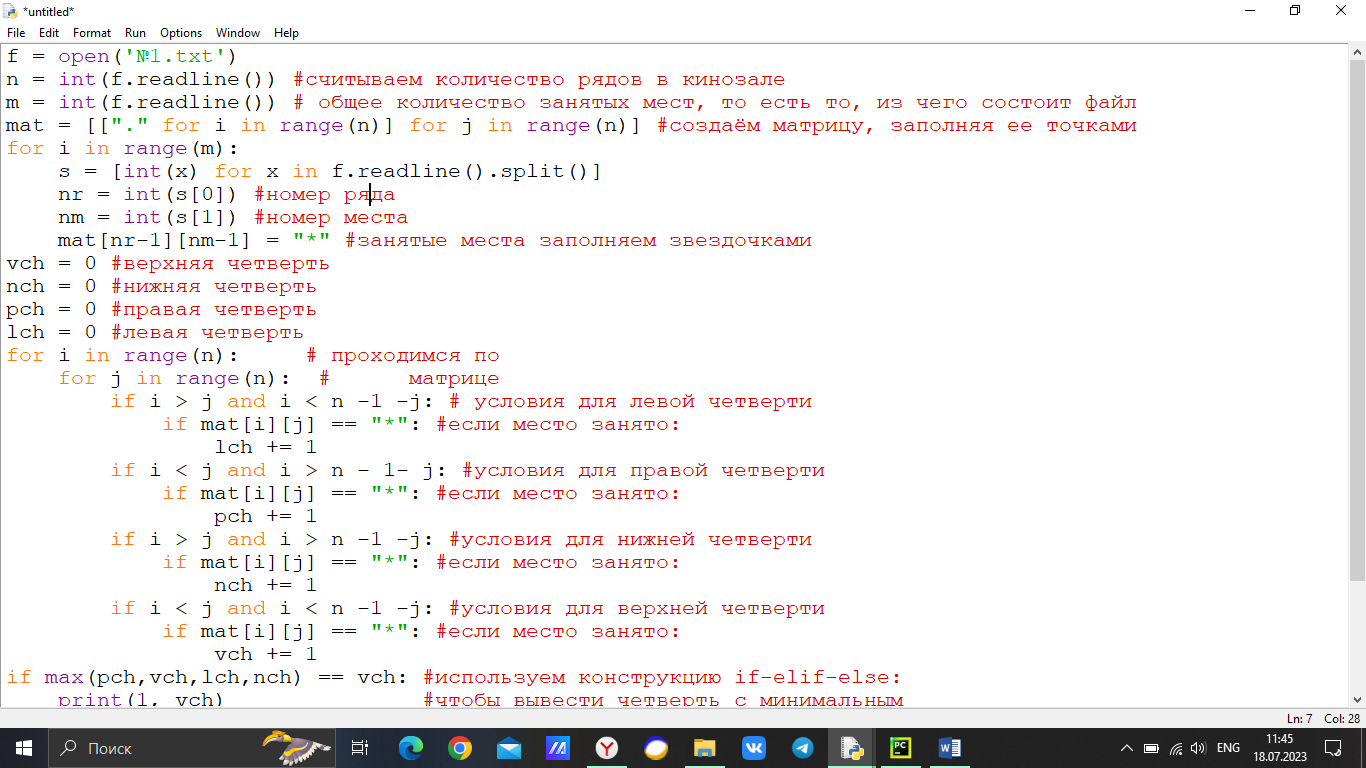
Решение: (программа)

1. Для начала считаем N и M, количество рядов в кинозале и общее количество занятых мест соответственно.
2. Для решения нам нужно создать матрицу (mat), чтобы наглядно отобразить ситуацию в зале. Создавая матрицу, заполним ее точками.
3. Проходимся по файлу циклом, считываем номер ряда и места и ставим на эти места в матрицу \*, таким образом в конце получим: «\*» - занятые места, «.» – пустые места.

**Примечание:** для того чтобы найти место и ряд, нам нужно вычитать единицу от номеров и ряда и места: C:\Users\DNS\Downloads\2023-07-18_11-05-07.pngЭто нужно для того чтобы мы не выходили за пределы, потому что индексы с 0, а номера с единицы.

1. Далее создаём переменные, которые будут отражать количество занятых мест в четвертях: 
2. Теперь один из самых сложных шагов. Нужно пройтись по матрице с помощью вложенных циклов, понять какая это четверть, и прибавить 1 к соответствующей переменной. Рассмотрим диагонали квадрата (зала). Элементы на диагональ с правого верхнего угла в левый нижний имеют индексы i = j, то есть их можно найти как mat[i] [i]. А вот элементы на диагонали с нижнего левого угла в правый верхний имеют такую зависимость в индексах: i = n – 1 – j, следовательно это mat[i][n-j-1]. То есть элементы в области, которая выше первой диагонали имеют i < j, а в той, которая ниже – i > j. Аналогично, элементы выше второй диагонали имеют индексы: i < n – 1- j, а которая выше – i > n – 1 – j. Объединив эти ограничения, получим: верхняя четверть находится выше и той, и другой диагонали, следовательно это все элементы, у которых индекс i < j and i < n-1-j, правая четверть находится выше первой и ниже второй диагоналей, следовательно i < j and I > n – 1- j. И так далее с остальными четвертями. Стоит заметить, что знаки “>”, ”<” строгие, потому что люди, сидящие на местах, расположенных на диагоналях, нас не интересуют (есть такое условие в задаче).
3. После того, как определили к какой четверти относится текущий элемент, стоит проверить, занято это место или нет условием: if mat[i][j] == “\*”, и только после этого стоит прибавлять единицу к соответствующей переменной, что будут означать количество занятых мест в каждой четверти по отдельности.
4. В конце программы нужно воспользоваться конструкцией if-elif-else, для того чтобы вывести четверть с минимальным номером, если интересующих нас четвертей одинаковое количество.

**Код Ниже (скриншот и текст)**



f = open('№1.txt')

n = int(f.readline()) #считываем количество рядов в кинозале

m = int(f.readline()) # общее количество занятых мест, то есть то, из чего состоит файл

mat = [["." for i in range(n)] for j in range(n)] #создаём матрицу, заполняя ее точками

for i in range(m):

s = [int(x) for x in f.readline().split()]

nr = int(s[0]) #номер ряда

nm = int(s[1]) #номер места

mat[nr-1][nm-1] = "\*" #занятые места заполняем звездочками

vch = 0 #верхняя четверть

nch = 0 #нижняя четверть

pch = 0 #правая четверть

lch = 0 #левая четверть

for i in range(n): # проходимся по

for j in range(n): # матрице

if i > j and i < n -1 -j: # условия для левой четверти

if mat[i][j] == "\*": #если место занято:

lch += 1

if i < j and i > n - 1- j: #условия для правой четверти

if mat[i][j] == "\*": #если место занято:

pch += 1

if i > j and i > n -1 -j: #условия для нижней четверти

if mat[i][j] == "\*": #если место занято:

nch += 1

if i < j and i < n -1 -j: #условия для верхней четверти

if mat[i][j] == "\*": #если место занято:

vch += 1

if max(pch,vch,lch,nch) == vch: #используем конструкцию if-elif-else:

print(1, vch) #чтобы вывести четверть с минимальным

elif max(pch,vch,lch,nch) == pch: #номером, если их несколько

print(2, pch)

elif max(pch,vch,lch,nch) == nch:

print(3, nch)

else:

print(4, lch)