5.1 task 7

Дано:

Входные параметры:



Задание:



Выходные параметры:

минимальный элемент кратный 3, среднеарифметическое отрицательных элементов четных строк и нечетных столбцов, а также элементы меньше его и их индексы

Дополнительные условия:



Текст программы:

from random import randint

def main():

#заполнение многомерного списка случайными элементами

def lt\_fill():

gridN = int(input('Количество строк: '))

gridM = int(input('Количество столбцов: '))

floor = int(input('Минимальный диапазон значений элементов: '))

ceiling = int(input('Максимальный диапазон значений элементов: '))

lt = [[randint(floor, ceiling) for i in range(gridM)]

for k in range(0, gridN)]

return lt

# очистить список

def lt\_clean(lt):

lt.clear()

# вывести список

def lt\_print(lt):

print(lt)

# найти минимальное значение кратное трем

def lt\_min\_3(lt):

minList = []

k = 0

for i in range(len(lt)): # поиск элемента

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

if lt[i][j] % 3 == 0:

minList.append(lt[i][j])

k += 1

if k != 0: # проверка наличия

print('Минимальный элемент массива кратный трем это: ', min(minList), '\n')

else:

print('Таких элементов нет\n')

#найти минимальное значение кратное пяти среди элементов с четными строками и нечетными столбцами

def lt\_max\_5(lt):

armid = int()

k = 0

for i in range(len(lt)): #проверка столбцов и строк

if i % 2 == 0:

for j in range(0, len(lt[i])):

if j % 2 != 0:

armid += lt[i][j] #сумма элементов

k += 1

armid /= k # среднее арифметическое

lesserList = []

lesserIndexList = [] # функциональные списки

k = 0

for i in range(len(lt)): # нахоэжение элементов меньше среднего арифметического с индексами

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

if lt[i][j] < armid and lt[i][j] < 0:

lesserList.append(lt[i][j])

lesserIndexList.append([i,j])

k += 1

if k > 0: # проверка наличия

print(f'Среднеарифметические отрицательных элементов массива равно {armid}, '

f'Элементы массива меньше его:')

for i in range(0, len(lesserList)):

if i % 5 == 0:

print()

print(lesserList[i], end=' ')

else:

print('Таких элементов нет')

# изменение элемента

def lt\_edit(lt):

i = int(input('Значение строки элемента (i): '))

j = int(input('Значение столбца элемента (j): '))

lt[i-1][j-1] = int(input('Новое значение элемента: '))

return lt

# пользовательский интерфейс

def menu():

ch = 1

while ch != 0:

ch = int(input('Выберите что вы хотите сделать:\n'

'1: Создать новый многомерный список \n'

'2: Очистить список \n'

'3: Вывести список \n'

'4: Найти минимальный элемент кратный трем \n'

'5: Найти среднеарифметическое, значения и индексы отрицательных элементов меньше его \n'

'6: Отредактировать элемент массива \n'

'0: Закрыть программу \n'))

if ch == 1:

lt = lt\_fill()

elif ch == 2:

lt\_clean(lt)

elif ch == 3:

lt\_print(lt)

elif ch == 4:

lt\_min\_3(lt)

elif ch == 5:

lt\_max\_5(lt)

elif ch == 6:

lt = lt\_edit(lt)

menu()

main()

Результат тестирования:





5.2 task 7

Дано:

Входные параметры:

Количество строк и столбцов массива

Задание:



Выходные параметры:

минимальный элемент кратный 3, среднеарифметическое отрицательных элементов четных строк и нечетных столбцов, а также элементы меньше его и их индексы

Дополнительные условия:



Текст программы:

from random import randint

def main():

# заполнение многомерного списка случайными элементами

def lt\_fill():

gridN = int(input('Количество строк: '))

gridM = int(input('Количество столбцов: '))

floor = int(input('Минимальный диапазон значений элементов: '))

ceiling = int(input('Максимальный диапазон значений элементов: '))

lt = [[randint(floor, ceiling) for i in range(gridM)]

for k in range(0, gridN)]

return lt

# очистить список

def lt\_clean(lt):

lt.clear()

# вывести список

def lt\_print(lt):

print(lt)

# функция проверки строки (или столбца) на наличие только положительных элементов

def lt\_check(lt):

k = 0

s = 0

for i in range(0, len(lt)):

for j in range(0, len(lt[i])):

if lt[i][j] > 0:

k += 1

if k == len(lt[i]):

s += 1

k = 0

if s > 0:

print('Имеется строка, состоящая только из положительных элементов')

else:

print('Такой строки нет')

s = 0

columnPositive = [] # функциональная строка

for i in range(0, len(lt)): #проверка индексов положительных элементов

for j in range(0, len(lt[i])):

if lt[i][j] > 0:

columnPositive.append(j)

length = len(lt[1])

for i in range(0, length): # проверка наличия положительного столбца

if columnPositive.count(i) == len(lt[i]):

s += 1

if s > 0:

print('Имеется столбец, состоящий только из положительных элементов')

else:

print('Такого столбца нет\n')

# пользовательский интерфейс

def menu():

ch = 1

while ch != 0:

ch = int(input('Выберите что вы хотите сделать:\n'

'1: Создать новый многомерный список \n'

'2: Очистить список \n'

'3: Вывести список \n'

'4: Проверить имеет ли строка или стобец только положительные элементы \n'

'0: Закрыть программу \n'))

if ch == 1:

lt = lt\_fill()

elif ch == 2:

lt\_clean(lt)

elif ch == 3:

lt\_print(lt)

elif ch == 4:

lt\_check(lt)

menu()

main()

Результат тестирования:



