**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**(Университет ИТМО)**

Факультет  **Инфокоммуникационных технологий**

Образовательная программа **Интеллектуальные системы в гуманитарной сфере**

Направление подготовки **45.03.04 Интеллектуальные системы в гуманитарной сфере**

О Т Ч Е Т

**лабораторной работе 4**

на тему: “Стек, очередь, связанный список”

Обучающийся (или несколько) ФИО, № группы

Королева Екатерина

К3143

Работа выполнена с оценкой **\_\_\_\_**

Преподаватель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Дата 08.10.2021

Санкт-Петербург, 2021

2. Очередь

from collections import deque

dq = deque()

answer = []

inp = open('input.txt')

out = open('output.txt', 'w')

n = int(inp.readline())

for i in range(n):

command = inp.readline()

if command[0] != "-":

command = int(command[1:])

dq.append(command)

else:

answer.append(dq.popleft())

out.write(str(answer))

print(answer)

inp.close()

out.close()

3. Скобочная последовательность. Версия 1

def is\_correct\_brackets(text):

if '()' in text or '[]' in text:

text = text.replace('()', '')

text = text.replace('[]', '')

return 'YES '

else:

return 'NO '

inp = open('input.txt')

out = open('output.txt', 'w')

n = int(inp.readline())

for i in range(n):

command = inp.readline()

out.write(str(is\_correct\_brackets(command)))

inp.close()

out.close()

5. Стек с максимумом

import random

def randomized\_quicksort(array, start, stop):

if start < stop:

key = random.randint(start, stop)

array[start], array[key] = array[key], array[start]

pivot = partition(array, start, stop)

randomized\_quicksort(array, start, pivot - 1)

randomized\_quicksort(array, pivot + 1, stop)

return array

def partition(array, start, stop):

b\_array = array[start]

c\_start = start

for i in range(start + 1, stop + 1):

if array[i] <= b\_array:

c\_start += 1

array[c\_start], array[i] = array[i], array[c\_start]

array[start], array[c\_start] = array[c\_start], array[start]

return c\_start

with open('input.txt', 'r') as f:

n, k = map(int, f.readline().split())

x = [[]] \* n

for i in range(n):

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

a = {}

for i in range(n):

a[float(((x[i][0] \*\* 2) + (x[i][1] \*\* 2)) \*\* 0.5)] = x[i]

list = [float(key) for key in a]

randomized\_quicksort(list, 0, len(list) - 1)

with open('output.txt', 'w') as f:

for i in range(k-1):

f.write(str(a[list[i]]))

f.write(',')

f.write(str(a[list[k - 1]]))

8. Постфиксная запись

def reverse\_polish(data):

stack = []

for i in data:

if i in '+\*-':

b = stack.pop()

print('b = ', b)

a = stack.pop()

print('a = ', a)

result = str(a)+i+str(b)

print('result = ', result)

stack.append(eval(result))

else:

stack.append(int(i))

print('i = ', i)

return stack

def output\_w(data\_):

text = ' '.join(map(str, data\_))

with open('output.txt', 'w') as f:

f.write(str(text))

inp = open('input.txt')

out = open('output.txt', 'w')

input\_data = inp.readlines()[1].split()

stack = reverse\_polish(input\_data)

output\_w(stack)