**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 #импортируем класс deque из модуля collections

dq = deque()#переменной присваиваем пустую очередь

answer = []#создаем пустой список

inp = open('input.txt')#открываем файл input

out = open('output.txt', 'w')#открываем файл output

n = int(inp.readline())#переменной присваем значение первой строки в файле input типа integer

for i in range(n):#заходим в цикл for

command = inp.readline()#переменной присвоить значение из файла input

if command[0] != "-":#если значение нулевого элемента не равно “-”, то:

command = int(command[1:])#переменной присвоить значение первого элемента типа integer

dq.append(command)#в очередь добавить переменную

else:#иначе

answer.append(dq.popleft())#в список 1 элемент из очереди

out.write(str(answer))#список положить в файл output

print(answer)#вывести список

inp.close()#закрыть файл input

out.close()#закрыть файл output

Время работы = 0,0006 секунд

3. Скобочная последовательность. Версия 1

def is\_correct\_brackets(text):#объявить функцию и подать на вход список

if '()' in text or '[]' in text:# если в списке верная скобочная последовательность, то

text = text.replace('()', '')#в списке заменить скобочную последовательность на пробел (дословно, убрать скобочную последовательность из списка)

text = text.replace('[]', '')#в списке заменить скобочную последовательность на пробел (дословно, убрать скобочную последовательность из списка)

return 'YES '#верни результат YES

else:#иначе

return 'NO '#верни результат NO

inp = open('input.txt')#открыть файл

out = open('output.txt', 'w')#открыть файл

n = int(inp.readline())#переменной присвоить первую строку из файла input типа integer

for i in range(n):#зайти в цикл

command = inp.readline()#переменной присвоить значение i строки

out.write(str(is\_correct\_brackets(command)))#в файл output вызвать функцию и положить результат ее работы

inp.close()#закрыть файл input

out.close()#закрыть файл output

Время работы = 0,0005 секунд

5. Стек с максимумом

import random# импортировать модуль random

def randomized\_quicksort(array, start, stop):#объявить функцию и дать на вход список и переменные start, stop

if start < stop:#если переменная start меньше переменной stop, то

key = random.randint(start, stop)#на

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

pivot = partition(array, start, stop)#найти индекс опорного элемента

randomized\_quicksort(array, start, pivot - 1)#вызвать функцию для первой половины списка

randomized\_quicksort(array, pivot + 1, stop)#вызвать функцию для второй половины списка

return array#вернуть список

def partition(array, start, stop):#объявить функцию и дать на вход список и переменные start, stop

b\_array = array[start]#переменной присвоить значение начального элемента списка

c\_start = start#переменной присвоить номер начального элемента списка

for i in range(start + 1, stop + 1):#зайти в цикл

if array[i] <= b\_array:#если итый элемент списка меньше либо равен элементу b\_array, то:

c\_start += 1#увеличить номер на 1

array[c\_start], array[i] = array[i], array[c\_start]#поменять мистами итый элемент списка и элемент с номером c\_start (внутри цикла)

array[start], array[c\_start] = array[c\_start], array[start]#поменять мистами итый элемент списка и элемент с номером c\_start (конечное)

return c\_start#вернуть номер элемента в списке

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

n, k = map(int, f.readline().split())#переменным присвоить последовательно первое и второе значение из первой строки файла в типе int

x = [[]] \* n#переменной присвоить значение элемента из списка списков, умноженное на n

for i in range(n):#зайти в цикл

x[i] = [int(j) for j in f.readline().split()]#итому элементу присвоить строку из файла

a = {}#создать пустой словарь

for i in range(n):#зайти в цикл

a[float(((x[i][0] \*\* 2) + (x[i][1] \*\* 2)) \*\* 0.5)] = x[i]#для каждого элемента в списке добавить значение в словарь по ключу

list = [float(key) for key in a]#в список положить значение ключа

randomized\_quicksort(list, 0, len(list) - 1)#вызвать функцию

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

for i in range(k-1):#зайти в цикл

f.write(str(a[list[i]]))#в файл положить итый элемент из списка в словаре типа string

f.write(',')#в файл положить запятушечку

f.write(str(a[list[k - 1]]))#в файл положить элемент из списка в словаре типа string

Время работы = 0,0005 секунд

8. Постфиксная запись

def reverse\_polish(data):#объявить функцию и дать на вход переменную

stack = []#создать пустой список

for i in data:#создать цикл

if i in '+\*-':#если итый элемент соответсвует

b = stack.pop()#запопим элемент b

a = stack.pop()#запопим элемент a

result = str(a)+i+str(b)#в переменную записать строку типа string вида число+число

stack.append(eval(result))#в список добавить переменную

else:#иначе

stack.append(int(i))#в список добавить итый номер типа integer

return stack#вернуть список

def output\_w(data\_):#объявить функцию и дать на вход переменную

text = ' '.join(map(str, data\_))#в список положить пробел и переменную типа string

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

f.write(str(text))#в файл положить список типа string

inp = open('input.txt')#открыть файл

out = open('output.txt', 'w')#открыть файл

input\_data = inp.readlines()[1].split()#переменной присвоить значение 1 строки файла input

stack = reverse\_polish(input\_data)#вызвать функцию

output\_w(stack)#в файл output положить результат работы функции

inp.close()#закрыть файл input

out.close()#закрыть файл output

Время работы = 0,007 секунд