Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

**Лабораторная работа № 4**

**по дисциплине**

**«Алгоритмы и структуры данных»**

**ДВОИЧНЫЕ КУЧИ**

**Выполнил**:

ст. гр. ПРИ-120

Д. А. Грачев

**Принял**:

Палик П.Г.

Владимир, 2021

ЦЕЛЬ РАБОТЫ

Разобраться с понятием двоичной кучи, научиться строить, объединять двоичные кучи, добавлять и удалять элементы.

ВЫПОЛНЕНИЕ РАБОТЫ

Для кучи реализовать следующие операции:

1. Добавить элемент;
2. Удалить элемент;
3. Построение кучи;
4. Объединение кучи.

Листинг программы:

start\_kucha = [100, 19, 36, 17, 3, 25, 1, 2, 7]

start\_kucha\_add = [16, 11, 9, 10, 5, 6, 8, 1, 2]

# def vivod\_kuchi(kucha):

#     for i in range()

def AddElem(kucha,flag=True,element=3456384):

    '''Добавляет элемент в кучу'''

    if flag:

        print (kucha)

        elem = int(input("Введите новый элемент: "))

    else:

        elem = element

    kucha.append(elem)

    index = len(kucha) - 1

    while kucha[index] > kucha[index//2]:

        kucha[index],kucha[index//2] = kucha[index//2],kucha[index]

        index //= 2

    if flag:

        print (kucha)

def DelElem(kucha):

    '''Удаляет больший элемент'''

    print (kucha)

    kucha[0] = kucha[-1]

    kucha.pop(-1)

    ind = 1

    n\_ind = ind \* 2

    while (n\_ind <= len(kucha)):

        if (n\_ind + 1 <= len(kucha)):

            if kucha[n\_ind-1]>kucha[n\_ind+1-1]:

                kucha[ind-1],kucha[n\_ind-1] = kucha[n\_ind-1],kucha[ind-1]

                ind = n\_ind

                n\_ind \*= 2

            else:

                kucha[ind-1],kucha[n\_ind] = kucha[n\_ind],kucha[ind-1]

                ind = n\_ind + 1

                n\_ind = (n\_ind + 1) \* 2

        else:

            kucha[ind-1],kucha[n\_ind-1] = kucha[n\_ind-1],kucha[ind-1]

    print (kucha)

def BuildKuch(flag=True,kucha=[]):

    if flag:

        dlina = int(input("Введите количество элементов: "))

        tmp\_kucha = []

        for i in range(dlina):

            vvod = int(input("Введите {}-й элемент: ".format(i+1)))

            tmp\_kucha.append(vvod)

    tmp\_kucha = kucha

    new\_kucha = []

    new\_kucha.append(tmp\_kucha[0])

    tmp\_kucha.pop(0)

    for elem in tmp\_kucha:

        AddElem(new\_kucha,False,elem)

    print (new\_kucha)

def MergeKuch(kucha1,kucha2):

    tmp\_kucha = kucha1 + kucha2

    BuildKuch(False,tmp\_kucha)

def menu():

    print ("Что вы хотите сделать?\n1 - Добавить элемент\n2 - Удалить больший эелемент\n3 - Построить кучу\n4 - Объединить кучи")

    key\_input = int(input("Выберите номер пункта: "))

    if key\_input == 1:

        AddElem(start\_kucha)

    elif key\_input == 2:

        DelElem(start\_kucha)

    elif key\_input == 3:

        BuildKuch()

    elif key\_input == 4:

        MergeKuch(start\_kucha,start\_kucha\_add)

menu()

Скриншот работы программы представлен на Рис. 1.

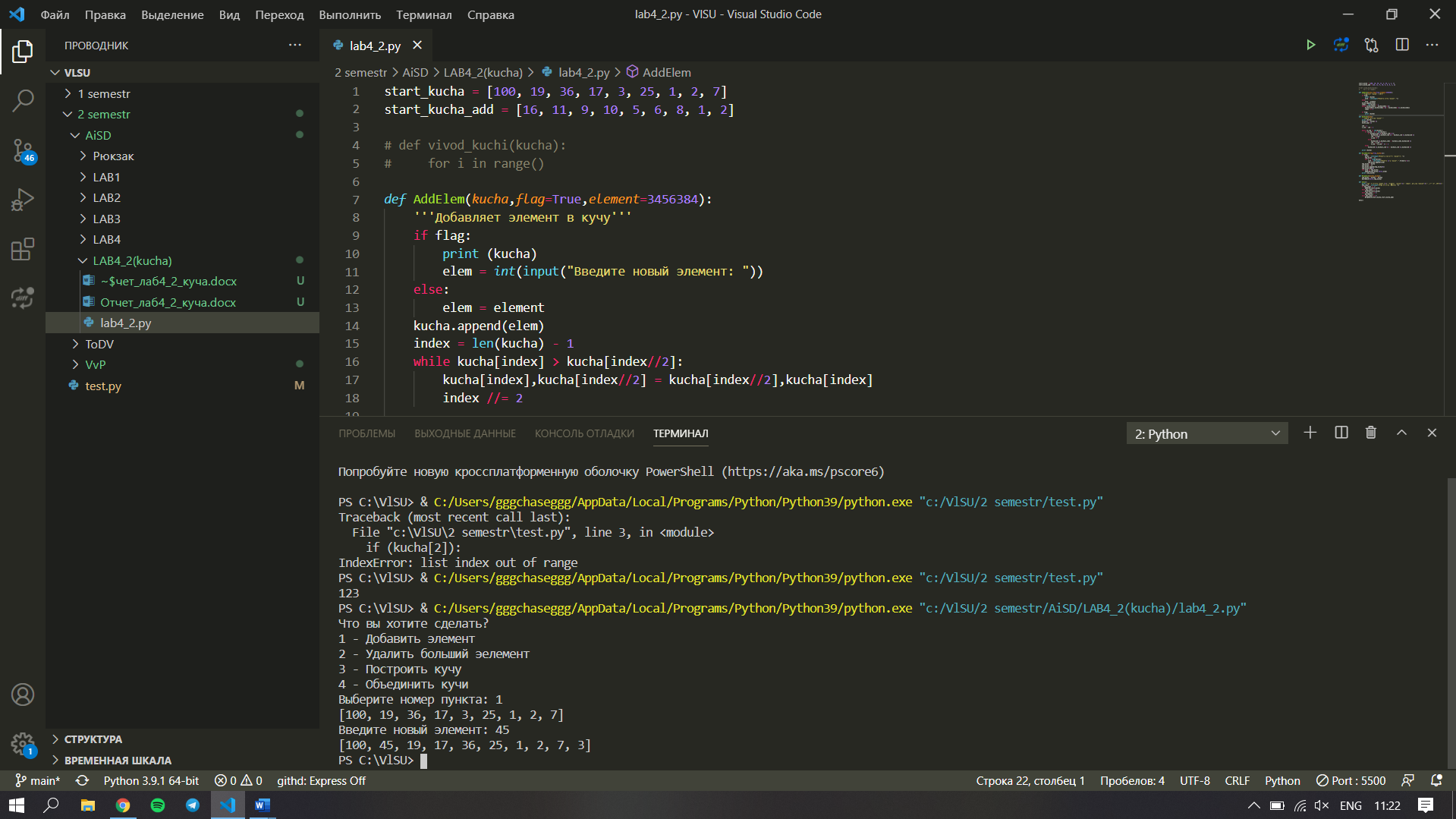


Рисунок 1. Скриншот работы программы

ВЫВОД

В ходе выполнения работы я разобрался с понятием двоичной кучи, научился строить, объединять двоичные кучи, добавлять и удалять элементы