Московский Авиационный Институт

(Национальный исследовательский университет)

**Институт №8 “Информационных технологий и прикладной математики”**

**Курсовой проект**

**по курсу**

**«Практикум программирования»**

**II семестр**

**Задание 9**

Студент: Калуцкий М.В.

Группа: М8О-107Б-22

Руководитель: Аносова Н.П.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя: \_\_\_\_\_\_\_\_\_\_\_\_

* 1. **Тема:** Сортировка и поиск.

1. **Цель работы:**  Составить программу на языке Си++ с использованием процедур и функций для сортировки таблицы заданным методом и двоичного поиска по ключу в таблице.
2. **Задание:** Составить программу, которая должна вводить значения элементов неупорядоченной таблицы и проверять работу процедуры сортировки в трех случаях: (1) элементы таблицы с самого начала упорядочены; (2) элементы таблицы расставлены в обратном порядке; (3) элементы таблицы не упорядочены. В последнем случае можно использовать встроенные процедуры генерации псевдослучайных чисел.

Для каждого вызова процедуры сортировки необходимо печатать исходное состояние таблицы и результаты сортировки. После выполнения сортировки программа должна вводить ключи и для каждого из них выполнять поиск в упорядоченной таблице с помощью процедуры двоичного поиска и печатать найденные элементы, если они присутствуют в таблице. Тестовые данные необходимо заранее поместить в текстовые файлы. В качестве текста для записей таблицы взять изображение ASCII-графики. Каждый элемент таблицы, содержащий ключ и текст записи, распечатывать в отдельной строке.

Метод сортировки – чётно-нечётная сортировка.

1. **Оборудование** (лабораторное):

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Intel Core i3-4005U CPU с ОП 8096 Мб, НМД \_ 131072\_ Мб. Монитор lenovo

1. **Программное обеспечение (лабораторное):**

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства \_\_\_Unix\_\_\_\_, наименование \_\_ Ubuntu \_\_\_\_\_\_\_\_\_\_\_\_ версия 4.4.2

интерпретатор команд \_\_\_\_bash\_\_\_\_\_\_\_\_ версия \_\_5.0.17\_\_\_\_.

Система программирования \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Редактор текстов \_\_\_\_\_nano\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ версия \_25.2.2 \_\_\_\_\_\_\_\_\_

Местонахождение и имена файлов программ и данных на домашнем компьютере \_\_\_\_\_home/jekich228\_\_\_\_\_\_\_\_\_

1. **Идея, метод, алгоритм** решения задачи(в формах:словесной,псевдокода,графической[блок-схема,диаграмма,рисунок,таблица] или формальные спецификации с пред- и постусловиями)

Для создания файла с перемешанными и развернутыми строками был реализован отдельный исполняемый файл “mix.cpp”. Для хранения ключа и соответствующей ему строки была создана структура “elem”. В данной структуре находится поле “key”, в котором хранится ключ строки, и поле “data”, в котором хранится сама строка. Сначала происходит считывание файла в вектор “picture”, который имеет тип данных <elem>, то есть вся картинка будет представлена в данном векторе в виде ключ – строка. Далее полученный вектор сортируется по ключам при помощи чётно-нечётной сортировки: суть чётно-нечётной сортировки заключается в том, что алгоритм проходит по массиву и сравнивает элементы на их нечётность или чётность. Он сортирует элементы, перемещая нечётные элементы в начало массива, а чётные элементы в конец.

Алгоритм начинает с фазы, называемой "нечётная фаза". В этой фазе он идёт по массиву, начиная с первого элемента, и сравнивает каждую пару элементов, стоящих рядом. Если текущий элемент и следующий элемент имеют разные чётности (нечётный и чётный), то они меняются местами. Таким образом, нечётные элементы "всплывают" к началу массива, а чётные элементы "опускаются" к его концу.

Затем следует фаза, называемая "чётная фаза". В этой фазе алгоритм выполняет аналогичные операции, но сравнивает и перемещает элементы, стоящие на чётных позициях массива. Это означает, что алгоритм сравнивает элементы с индексами 0 и 2, затем 2 и 4, и так далее.

1. **Сценарий выполнения работы** [план работы,первоначальный текст программы в черновике(можно на отдельном листе)итесты либо соображения по тестированию].

**Файл “main.cpp”:**

#include "binary\_search.hpp"

int main(){

    wf();

    string e;

    while (1)

    {

        cin>>e;

        if(e == "exit")break;

        int t = binary\_search(stoi(e));

        if(t==-1){cout<<"element not found";}

        else cout<<picture[t].data<<endl;

    }

}

**Файл “mix.cpp”:**

#include<bits/stdc++.h>

using namespace std;

ifstream in("text3.txt");

int c = 29;

struct elem

{

    int key;

    string data;

};

vector<elem> picture;

int main()

{

    std::random\_device rd;

    std::mt19937 gen(rd());

    std::uniform\_int\_distribution<> dist(1, 29);

    std::set<int> unique\_numbers;

    vector<int> a;

    while(a.size() < 29)

    {

        int i = dist(gen);

        int s = unique\_numbers.size();

        unique\_numbers.insert(i);

        if(s!=unique\_numbers.size()){

            a.push\_back(i);

        }

    }

   for(int i = 0; i < c; ++i){

        elem kek;

        in >> kek.key;

        in >> kek.data;

        picture.push\_back(kek);

    }

    ofstream file1("text1.txt");

    for(int i:a){

        file1<<picture[i-1].key<<picture[i-1].data<<endl;

    }

    file1.close();

    ofstream file2("text2.txt");

    for(int i = 29;i>0;--i){

        file2<<picture[i-1].key<<picture[i-1].data<<endl;

    }

    file2.close();

}

**Файл “binary\_search.hpp”:**

#include "reading.hpp"

int binary\_search(int e){

    int low = 0, high = 28, middle;

    while(low <= high){

        middle = (high+low)/2;

        if(e>middle)low = middle+1;

        else if(e<middle) high = middle-1;

        else return middle;

    }

    return -1;

}

**Файл “reading.hpp”:**

#include<bits/stdc++.h>

using namespace std;

ifstream in("text1.txt");

int c = 29;

struct elem

{

    int key;

    string data;

};

vector<elem> picture;

void wf(){

    for(int i = 0; i < c; ++i){

        elem kek;

        in >> kek.key;

        in >> kek.data;

        picture.push\_back(kek);

    }

    for(elem i : picture){

        if(i.key<10){

            cout<<i.key<<"  "<<i.data<<endl;

        }

        else cout<<i.key<<" "<<i.data<<endl;

    }

    cout<<endl;

    cout<<endl;

    cout<<endl;

    int k = 0;

    while(1){

        k = k%2;

        int ch = 2\*((picture.size()-k)/2);

        int y = 0;

        for(int i = k;i<ch;i=i+2){

            if(picture[i].key > picture[i+1].key){

                elem t = picture[i];

                picture[i] = picture[i+1];

                picture[i+1] = t;

                y = 1;

            }

        }

        if(y==0) break;

        k++;

    }

    for(elem i : picture){

        if(i.key<10){

            cout<<i.key<<"  "<<i.data<<endl;

        }

        else cout<<i.key<<" "<<i.data<<endl;

    }

    in.close();

    cout<<endl;

    cout<<endl;

}

**Файл “text1.txt”:**

16⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

28⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

20⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

21⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

15⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

29⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

19⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

17⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

8⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

2⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

24⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

5⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

10⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

4⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

18⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

1⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

12⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

23⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

22⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

9⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

7⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

26⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

27⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

13⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

6⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

3⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

11⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

14⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

25⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

**Файл “text2.txt”:**

29⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

28⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

27⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

26⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

25⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

24⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

23⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

22⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

21⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

20⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

19⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

18⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

17⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

16⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

15⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

14⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

13⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

12⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

11⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

10⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

9⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

8⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

7⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

6⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

5⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

4⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

3⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

2⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

1⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

**Файл “text3.txt”:**

1 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

2 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

3 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

4 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

5 ⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

6 ⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

7 ⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

8 ⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

9 ⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

10 ⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

11 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

12 ⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

13 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

14 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

15 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

16 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

17 ⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

18 ⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

19 ⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

20 ⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

21 ⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

22 ⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

23 ⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

24 ⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

25 ⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

26 ⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

27 ⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

28 ⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

29 ⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

1. **Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами,подписанныйпреподавателем).

PS C:\Users\heleu\Desktop\laba\_2\_sem\_kek> & 'c:\Users\heleu\.vscode\extensions\ms-vscode.cpptools-1.15.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-rpere3mg.ldu' '--stdout=Microsoft-MIEngine-Out-c1q2ythg.4bk' '--stderr=Microsoft-MIEngine-Error-gj4xuxn1.i2i' '--pid=Microsoft-MIEngine-Pid-t4a2sg2h.mte' '--dbgExe=C:\MinGW\bin\gdb.exe' '--interpreter=mi'

16 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

28 ⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

20 ⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

21 ⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

15 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

29 ⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

19 ⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

17 ⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

8 ⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

2 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

24 ⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

5 ⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

10 ⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

4 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

18 ⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

1 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

12 ⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

23 ⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

22 ⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

9 ⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

7 ⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

26 ⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

27 ⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

13 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

6 ⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

3 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

11 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

14 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

25 ⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

1 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

2 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

3 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

4 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

5 ⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

6 ⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

7 ⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

8 ⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

9 ⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

10 ⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

11 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

12 ⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

13 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

14 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

15 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

16 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

17 ⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

18 ⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

19 ⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

20 ⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

21 ⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

22 ⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

23 ⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

24 ⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

25 ⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

26 ⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

27 ⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

28 ⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

29 ⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

12

⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

1

⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

2

⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

3

⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

4

⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

5

⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

6

⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

exit

PS C:\Users\heleu\Desktop\laba\_2\_sem\_kek>

PS C:\Users\heleu\Desktop\laba\_2\_sem\_kek> & 'c:\Users\heleu\.vscode\extensions\ms-vscode.cpptools-1.15.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-xjappgx2.o22' '--stdout=Microsoft-MIEngine-Out-rkxg5t5d.xa4' '--stderr=Microsoft-MIEngine-Error-nmkolmer.rbv' '--pid=Microsoft-MIEngine-Pid-a1aqbdlt.1tz' '--dbgExe=C:\MinGW\bin\gdb.exe' '--interpreter=mi'

29 ⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

28 ⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

27 ⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

26 ⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

25 ⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

24 ⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

23 ⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

22 ⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

21 ⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

20 ⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

19 ⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

18 ⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

17 ⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

16 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

15 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

14 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

13 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

12 ⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

11 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

10 ⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

9 ⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

8 ⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

7 ⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

6 ⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

5 ⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

4 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

3 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

2 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

1 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

1 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

2 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡿⣿⡟⠁⠹⣿⣿⣿⣿⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

3 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⡏⠄⡿⠄⣠⠄⠈⠙⠻⢿⣿⠄⠙⢿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

4 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⠿⣛⠛⠓⠄⢀⠄⣿⣄⡀⠄⡀⠄⢻⠄⣤⠄⠛⠉⣼⣿⢿⣿⣿⣿⣿⣿⣿⣿⣿

5 ⣿⣿⣿⣿⣿⣿⡟⢿⣿⣿⡿⠛⠻⣂⣤⣼⣧⣿⣿⣿⣿⣾⣷⣄⣼⣿⣇⢀⠄⢋⠁⠘⣿⣿⣿⣿⣿⣿⣿⣿

6 ⣿⣿⣿⣿⣿⣿⣇⠄⠈⢉⡁⠄⠚⢛⣻⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣯⣴⣿⡇⣰⣿⣿⣿⣿⣿⣿⣿⣿

7 ⣿⣿⣿⣿⣿⣿⡄⠄⠄⠈⣿⣶⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣉⡁⢀⣼⣿⣿⣿⣿⣿⣿

8 ⣿⣿⣿⣿⣿⣿⠿⠂⠄⠄⢻⣿⣿⣿⣿⡿⠛⣻⣿⣿⣿⣿⣿⡏⢻⣿⣿⣿⣿⣿⣟⠉⠄⠙⠻⠿⢻⣿⣿⣿

9 ⣿⣿⣿⣿⣿⣿⣶⠆⠄⠄⣻⣿⣿⣿⣏⣴⠶⢟⣽⣿⣿⣿⣿⣷⣿⣿⣿⣿⣿⣿⣷⣦⣴⣶⠄⢖⣿⣿⣿⣿

10 ⣿⣿⣿⣧⣍⠉⢤⣤⣤⣾⡿⣿⣿⣿⣿⡇⢀⣾⡿⢻⣿⣿⣿⣿⣿⣿⣿⣿⣷⢹⣿⣿⣿⠋⣠⣾⣿⣿⣿⣿

11 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣼⣿⣿⣿⡿⡇⣼⠏⠄⢸⣿⣿⡿⣿⡿⠹⣿⣿⣿⣿⣿⣿⣿⣷⡈⠙⢿⣿⣿⣿

12 ⣿⣿⣿⣿⣿⣏⣰⣿⣿⣿⣿⣿⡿⣿⠵⠷⣧⡀⠠⠿⠟⠋⢀⣸⠷⠾⣝⣿⣿⣿⣿⣿⣿⢦⣶⣶⣾⣿⣿⣿

13 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⢙⣿⣿⣼⠁⠄⢠⣿⡏⠏⠄⠄⠉⢻⣿⡆⠄⠼⢿⣿⣿⡿⣿⣿⣿⣿⣿⣿⣿⣿⣿

14 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⡛⡿⠽⡀⠄⠄⠙⠁⠄⠄⠄⠄⠈⠛⠁⠄⢀⡞⢫⢟⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

15 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣷⠠⠚⡑⠒⠉⠄⠄⠄⠄⠄⠄⠈⠐⠪⠙⡀⣾⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

16 ⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣧⡁⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣼⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿

17 ⣿⣿⣿⣿⣿⣿⣿⣿⡿⠟⢋⣉⣤⣽⣆⡀⠄⠄⠒⠒⠒⠒⠂⠄⢀⣠⣾⣤⣌⣉⠛⠻⣿⣿⣿⣿⣿⣿⣿⣿

18 ⣿⣿⣿⣿⣿⣿⣿⡟⠄⣶⣿⣿⣿⣿⣧⠙⢶⣄⡀⠄⠄⢀⣠⡴⡿⣸⣿⣿⣿⣿⣷⡆⠈⣿⣿⣿⣿⣿⣿⣿

19 ⣿⣿⣿⣿⣿⡿⢛⠁⠄⠙⢿⣿⣿⣿⠿⡆⠄⠈⠻⣷⣾⡿⠋⢸⠃⡿⣿⣿⣿⣿⠟⠃⠄⢙⠻⣿⣿⣿⣿⣿

20 ⣿⣿⣿⣿⡿⠁⢿⣧⡀⠄⠄⠙⠻⣅⠄⠄⠄⠄⠄⢸⣿⠁⠄⠸⠄⠁⢀⡽⠛⠁⠄⠄⣠⣾⡇⠸⣿⣿⣿⣿

21 ⣿⣿⣿⣿⡇⠄⠄⠉⠙⠢⣄⠄⠄⠄⡇⠄⠄⢀⠄⠈⠋⠄⠄⠄⠄⢠⠎⠄⠄⢀⡤⠞⠋⠁⠄⠄⢻⣿⣿⣿

22 ⣿⣿⣿⣿⠁⠄⠄⠄⠄⠄⠄⠙⠢⣄⣇⠄⠄⠈⡂⠄⠄⢀⠄⠄⡀⡏⢀⡠⠞⠉⠄⠄⠄⠄⠄⠄⢸⣿⣿⣿

23 ⣿⣿⣿⠟⠉⠙⠷⣦⣀⠄⠄⠄⠄⠈⢫⠄⠈⠒⠴⠄⢀⠧⠔⠉⢸⠑⠋⠄⠄⠄⠄⠄⣠⣴⠟⠉⠙⠻⣿⣿

24 ⣿⣿⠃⠄⠄⠄⠄⠉⢿⣿⣶⣦⣤⣄⣀⠳⢄⡀⠄⠄⠄⠄⣀⣴⠃⢀⣀⣤⣤⣶⣾⣿⠟⠁⠄⠄⠄⠄⠹⣿

25 ⣿⣧⠄⠄⠄⠄⠄⠄⠄⠈⠛⢿⣿⣿⣿⣿⣾⠙⠲⠤⠔⠊⠁⢸⣾⣿⣿⣿⣿⠿⠋⠁⠄⠄⠄⠄⠄⠄⢀⢿

26 ⣿⢹⠄⠄⠄⡤⠄⠄⠄⠄⠄⠄⠄⠉⠛⣿⣿⠄⠄⠄⠄⠄⠄⢸⣿⡿⠛⠉⠄⠄⠄⠄⠄⠄⠄⡄⠄⠄⢸⢸

27 ⡏⠘⡇⠄⣰⡇⠄⠄⠄⠄⠄⠄⠄⠄⠄⠈⣿⠄⠄⠄⠄⠄⠄⢸⡟⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢿⡀⠄⢸⠄

28 ⠁⠄⡇⢀⣿⠃⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⠄⠄⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣷⠄⡟⠄

29 ⠄⠄⢳⣸⣿⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⣿⠄⠄⠄⠄⣠⠎⢸⢸⠄⠄⠄⠄⠄⠄⠄⠄⠄⠄⢸⣿⣰⠇⠄

exit

PS C:\Users\heleu\Desktop\laba\_2\_sem\_kek>

1. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные события(ошибки в сценарии и программе,нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  | или |  |  |  |  |  |
|  | дом. |  |  |  |  |  |
|  |  |  |  |  |  |  |

1. **Замечания автора** по существу работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Выводы**: Проделав данную работу, я научился составлять программы на языке Си++ с использованием процедур и функций для сортировки таблицы заданным методом и двоичного поиска по ключу в таблице.

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Идея, метод, алгоритм** решения задачи(в формах:словесной,псевдокода,графической[блок-схема,диаграмма,рисунок,таблица] или формальные спецификации с пред- и постусловиями)
2. **Сценарий выполнения работы** [план работы,первоначальный текст программы в черновике(можно на отдельном листе)итесты либо соображения по тестированию].
3. **Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами,подписанныйпреподавателем).
4. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные события(ошибки в сценарии и программе,нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  | или |  |  |  |  |  |
|  | дом. |  |  |  |  |  |
|  |  |  |  |  |  |  |

1. **Замечания автора** по существу работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Выводы**: Проделав данную работу, я научился составлять программы выполнения заданных действий над файлами на интерпретируемом командном языке Bash

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_