

Институт математики, естественных и компьютерных наук

Кафедра автоматики и вычислительной техники

## Отчет по лабораторной работе №2

Дисциплина: «Программирование»

<b>09.03.04</b>	<b>43.10</b>		<b>01</b>	<b>2025</b>
Код направления подготовки/ специальности	Код выпускающей кафедры	Регистрационный номер по журналу	Код формы обучения	Год

Руководитель доц., Кочкин Дмитрий Валерьевич

Выполнил студент Тестов Максим Олегович

Группа, курс 4Б09 РПС-21

Дата сдачи \_\_\_\_\_

Дата защиты \_\_\_\_\_

Оценка по защите \_\_\_\_\_

## 1. Сортировка символов

Напишите программу, сортирующую символы строки, введенные с клавиатуры (в строке содержатся ТОЛЬКО английские символы; заглавные буквы имеют больший приоритет, чем прописные, т.е. заглавные буквы должны находиться в начале строки), в алфавитном порядке. Например, если введено: "CbA в результате надо получить "ACb".

### Входные данные

Строка с английскими символами (до 255 символов).

### Выходные данные

Строка с отсортированным порядком символов

### Пример входные данные

ajJGFghHfGHGHffsdgfgshdgfsdgfhsgdgh

### Пример выходные данные

FGGGHHHJadddddffffffffffggggggghhhhjsssss

### Решение:

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string str;
    cin >> str;
    for(int i = 0; i < str.length(); i++) {
        for (int j = 0; j < str.length() - i - 1; j++) {
            if (str[j] > str[j + 1]) {

                char k = str[j];
                str[j] = str[j + 1];
                str[j + 1] = k;
            }
        }
    }
    cout << str << endl;
    return 0;
}
```

## 2. Спиральный массив

Заполнение массива 6\*6 по спирали. Вывести на экран. Начать заполнение с первого элемента первой строки матрицы. На рисунке ниже показано, в каком порядке нужно заполнять матрицу 6\*6 (стрелками показано направление обхода). 36 элементов вводятся с клавиатуры (каждый элемент в отдельной строке). Полученный массив-спираль вывести на экран разделяя строки переносом на другую строку, а элементы в строках - знаком табуляции.

```
- - -> - - +
+ - -> - + |
| + ->+ | |
| | <-+ | |
| + -<- + |
+ - -<- - +
```

### Исходные данные

36 элементов массива вводятся с клавиатуры (каждый элемент в отдельной строке)

### Результат

Массив-спираль вывести на экран разделяя строки переносом на другую строку, а элементы в строках - знаком табуляции.

### Решение:

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string spiral[6][6];
    for(int j = 0; j < 6 ; j++){
        cin >> spiral[0][j];
    }
    for(int j = 1; j < 6 ; j++){
        cin >> spiral[j][5];
    }
    for(int j = 4; j >= 0 ; j--){
        cin >> spiral[5][j];
    }
    for(int j = 4; j > 0 ; j--){
        cin >> spiral[j][0];
    }
    for(int j = 1; j < 5 ; j++){
        cin >> spiral[1][j];
    }
}
```

```

for(int j = 2; j < 5 ; j++){
    cin >> spiral[j][4];
}
for(int j = 3; j >= 1 ; j--){
    cin >> spiral[4][j];
}
for(int j = 3; j > 1 ; j--){
    cin >> spiral[j][1];
}
for(int j = 2; j < 4 ; j++){
    cin >> spiral[2][j];
}
for(int j = 3; j > 1 ; j--){
    cin >> spiral[3][j];
}

for(int i = 0; i < 6 ; i++){

    for(int j = 0; j < 6 ; j++){
        cout << spiral[i][j] <<"\t";
    }
    cout <<"\n";
}

return 0;
}

```

### 3. Удаление k-го символа

Напишите программу, которая вводит строку и выводит ее, сокращая каждый раз на K-ый символ до тех пор (вывод на экран после каждого сокращения), пока в строке не останется K-1 символов. Символы в строке нумеруются с 1. Если K больше длины строки, то ничего не выводить на экран.

#### Входные данные

Строка текста (до 255 символов).

Целое число K (от 1 до 1000).

#### Выходные данные

Строки согласно условию.

#### Пример входные данные

1234567890

3

#### Пример выходные данные

124567890

12567890

1267890

127890

12890

1290

120

12

#### Решение:

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string str;
    int k;
    getline(cin, str);
    cin >> k;
    cout << endl;
    if((str.length()) >= k && (k > 0)){
        while(str.length() > k - 1){
            str.erase(k-1, 1);
            cout << str << endl;
        }
    }
    return 0;
}
```