

**Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
"Уфимский государственный авиационный технический университет"**

Кафедра Высокопроизводительных вычислительных технологий и систем

Дисциплина: Программирование

Отчет по лабораторной работе № 4

Тема:

Группа ПМ-153	Фамилия И.О.	Подпись	Дата	Оценка
Студент	Шамаев И.Р.			
Принял	Гайнетдинова А.А.			

Уфа 2019

Цель: изучить принципы описания строки и работы ними в СИ, функции для работы со строками, а также организацию ввода-вывода с использованием файлов.

Теоретический материал

Осуществили работу с файлами, открывали их в разных режимах, для чтения и для редактирования

Ответы на контрольные вопросы

Как описывается строка в Си? Чем является строка в Си?

Строка — последовательность символов. Для работы с символами в языке C++ предусмотрен тип данных `char`. Если в выражении встречается одиночный символ, он должен быть заключён в одинарные кавычки, например, `'a'` или `'*'`. Последовательность символов, то есть строка, заключается в двойные кавычки. Признаком конца строки является нулевой символ `'\0'`. В Си строки можно описать с помощью массива символов (массив элементов типа `char`), в массиве следует предусмотреть место для хранения признака конца строки (`'\0'`), то есть размер массива будет на единицу больше, чем количество символов.

В какой библиотеке находятся основные функции для работы со строками?

```
#include <string.h>
```

Как сравниваются строки в Си? Как задается значение строковой переменной в Си?

`strcmp` – сравнение строк

```
Char s[]='пример'
```

Какие функции используются для ввода строк в Си? Чем они отличаются?

`Scanf()`

`Gets()` - Считывает символы из стандартного ввода (`stdin`) и сохраняет их в виде строки `C` в `str` до тех пор, пока не будет достигнут символ новой строки или конец файла.

`Fgets()` - Он читает строку из указанного потока и сохраняет ее в строке, на которую указывает `str`. Он останавливается, когда читается (n-1) символов, читается символ новой строки или достигается конец файла, в зависимости от того, что произойдет раньше.

С какими типами файлов можно работать в программе, написанной на Си?

Текстовые и бинарные. Текстовые – последовательность символов, бинарные последовательность данных

Опишите прототипы функций для создания/открытия и закрытия файлов.

foopen() - открыть файл

fclose() - закрыть файл

Какие опции (флаги) используются в функции для открытия файла.

"r" - Открытие существующего файла только для чтения

"w" - Открытие файла только для записи. Если файл уже существует его содержимое очищается. В противном случае будет создан новый файл

"a" - Открытие файла для добавления, т.е. только запись в конец файла.

Если файл уже существует его содержимое не изменяется, а новые данные дописываются в конец. В противном случае создается новый файл.

"r+" - Открытие существующего файла как для записи так и для чтения.

Содержимое файла не изменяется, позиция чтения/записи установлена в начало файла. При записи старое содержимое будет заменяться на новое.

"w+" - Открытие существующего файла как для записи так и для чтения.

Если файл уже существует его содержимое очищается. В противном случае будет создан новый файл

"a+" - Открытие файла как для чтения так и для добавления. Если файл

существует — его содержимое не изменяется, позиция чтения

установлена на начало файла, добавление выполняется в конец

файла/записи установлена в начало файла. Если файла не существует — он создается.

Опишите функции для ввода/вывода информации в/из файла.

FSCANF()

FPRINTF()

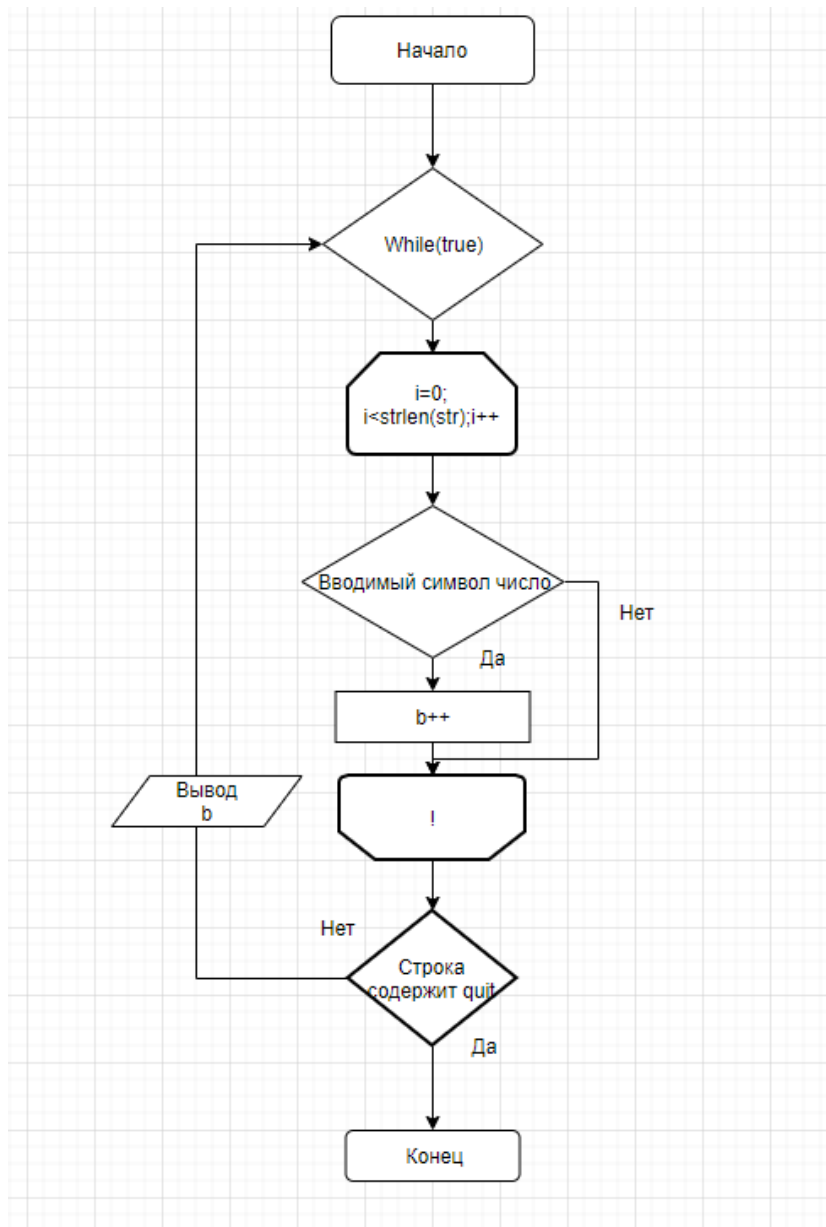
FPUTC() – Функция записывает символ в поток и перемещает позицию индикатора положения.

FPUTS() - Функция записывает строку, указанную в параметре string

Индивидуальное задание №1

Задание: Получить от пользователя строку текста и Определить количество знаков, не являющихся буквами цифрами. Повторять до тех пор, пока не встретится строка завершающаяся словом quit

Блок-схема



Описание программы:

b – счетчик чисел в введенной строке

j- счетчик, который проверяет, было ли введено слово quit

Вводим строку, проверяем сколько символов в строке, не являющихся буквами и цифрами. Проверяем было ли введено слово quit. В зависимости от этого либо выходим из программы, либо выводим количество символов

Исходный код программы

```
#include <stdio.h>
#include <ctype.h>
#include <string.h>

using namespace std;

int main()
{   int b=0, j=0;
    char str[15];

    while(true){

        printf("\nEnter string: ");
        gets(str); // Вводим строку

        for(int i=0; i < strlen(str); i++){
            if (!isalpha(str[i]) && !isdigit(str[i])){//проверяем, сколько символов в
строке не являются буквами и цифрами
                b++; //счетчик символов
            }

            if((str[strlen(str)-5] == ' ')&&(str[strlen(str)-4] == 'q')&&(str[strlen(str)-
3] == 'u')&&(str[strlen(str)-2] == 'i')&&(str[strlen(str)-1] == 't')&&(str[strlen(str)]
== '\0')){
                j++;
            }
        }

        if(j!=0){ // было ли введено слово quit
            return 0;
        }

        printf("%d", b); //вывод количества чисел в введенной строке
        b=0;

    }

    return 0;
```

}

Пример выполнения программы

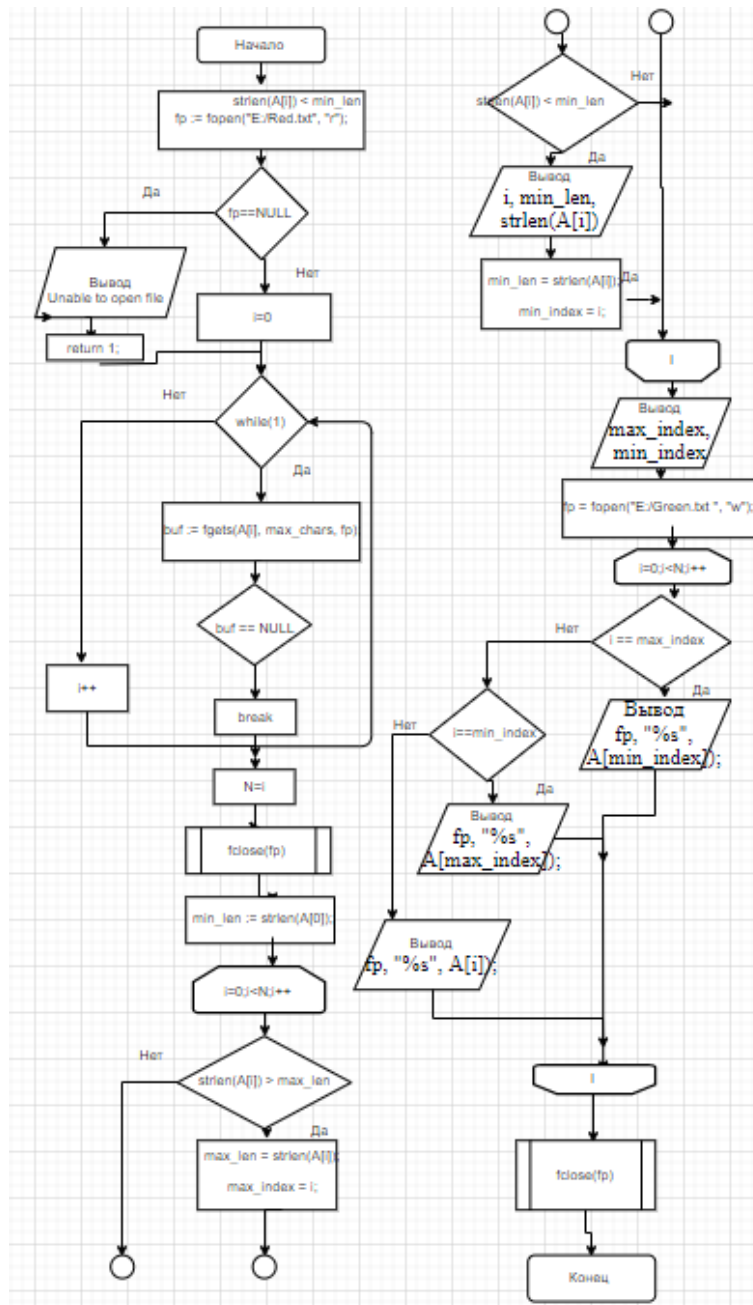
```
Enter string: gsgdgasd quitg
1
Enter string: quit fgdgdfgdfg
2
Enter string: dshsdfs dsfhhsdhf quit

Process returned 0 (0x0)   execution time : 15.265 s
Press any key to continue.
```

Индивидуальное задание №2

Задание: Открыть текстовый файл в необходимом режиме (на чтение, на чтение и запись, на добавление). (В файле должно быть не менее 30 строк). Поменять местами самую длинную и самую короткую строки текста. Записать измененный текст в новый файл.

Блок-схема



Описание программы:

Исходный код программы

```

#include <stdio.h>
#include <string.h>

int main(){
    FILE *fp;
    int i, j, N, max_chars=255, min_index, max_index, min_len=0, max_len=0;
    char A[30][255];
    char* buf;

    fp = fopen("E:/Red.txt", "r");
    if (fp == NULL){
        printf("Unable to open file");
        return 1;
    }
    i = 0;
    while (1)
    {
        buf = fgets(A[i], max_chars, fp);
        if (buf == NULL)
        {
            break;
        }
        i++;
    }
    N = i;
    fclose(fp);
    min_len = strlen(A[0]); // записываем длину первой строки как
минимальную
    for (i=0; i<N; i++)
    {
        if (strlen(A[i]) > max_len){ // если длина текущей больше чем
максимальная длина
            max_len = strlen(A[i]);
            max_index = i;
        }
        else {
            if (strlen(A[i]) < min_len){ // если длина текущей меньше чем
минимальная

```



```

        printf("else, i = %d, min_len = %d, strlen = %d\n", i, min_len,
strlen(A[i]));
        min_len = strlen(A[i]);
        min_index = i;
    }
}
}
printf("max_index = %d, min_index = %d", max_index, min_index);
fp = fopen("E:/Green.txt ", "w");
for (i=0; i<N; i++){
    if (i == max_index){ //меняем местами мин и макс строчки
        fprintf(fp, "%s", A[min_index]);
    } else {
        if (i==min_index){
            fprintf(fp, "%s", A[max_index]);
        } else {
            fprintf(fp, "%s", A[i]);
        }
    }
}
fclose(fp);
}

```

Пример выполнения программы

Red - Блокнот	*Green - Блокнот	"E:\CodeBlocks\share\CodeBlocks\--ютр яряър\df\bin\Debug\df.exe"
Файл Правка Фо	Файл Правка Фо	else, i = 1, min_len = 9, strlen = 4
34234234	34234234	max_index = 3, min_index = 1
979	9880234888	Process returned 0 (0x0) execution time : 0.042 s
1231231	1231231	Press any key to continue.
9880234888	979	

Индивидуальное задание №3

Задание: Описать структуру с именем PLANET, содержащую следующие поля:

NAME – название планеты;

RADIUS_PL – Радиус планеты;

RADIUS_ORB – Радиус орбиты;

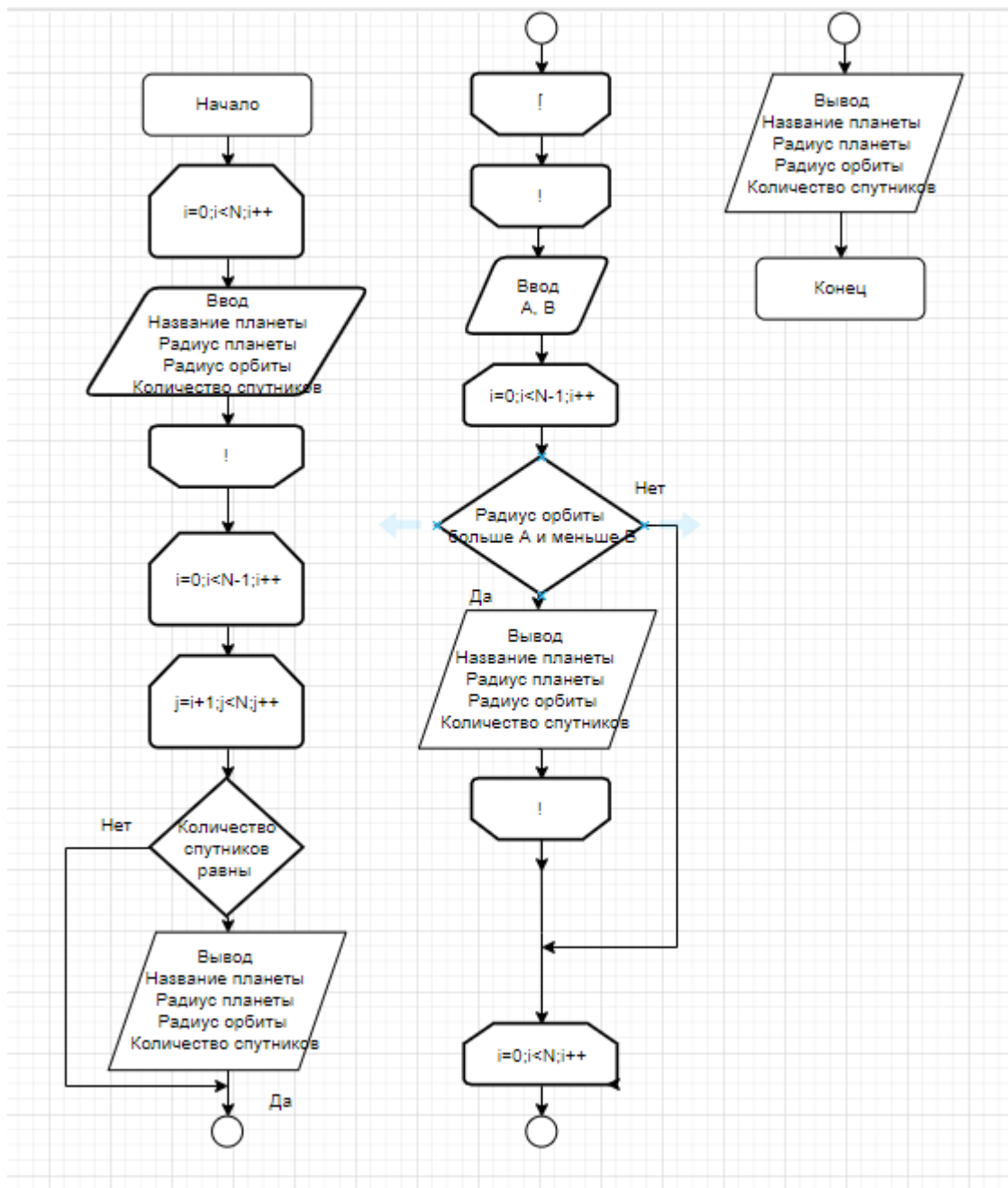
SATELLITES_QNT – количество спутников, как естественных, так и искусственных;

SAT_NAMES – массив с названиями спутников, как естественных, так и искусственных;

Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив STAR_SYSTEM из N элементов типа PLANET, описывающий состав звездной системы; (ввести не менее 5 планет с указанием спутников как минимум для трех из них)
- вывод на экран информации о всех планетах с одинаковым количеством спутников;
- вывод на экран информации о планетах, радиус орбиты которых находится в указанном пользователем диапазоне. Если в указанной диапазоне нет планет, то выводится информация о всей звездной системе

Блок-схема



Описание программы:

Описываем структуру. Вводим 5 планет и данные для каждой из них.

1)Проверяем равны ли количество спутников, и выводим данные планет, у которых количество спутников совпадают

2)Проверяем равны ли радиусы орбит, и выводим данные планет, у которых радиусы орбит равны, если таких нет, выводим все данные по всем введенным планетам

Исходный код программы

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```

#include <string.h>
using namespace std;

struct PLANET {
    char NAME[30];
    int RADIUS_PL;
    int RADIUS_ORB;
    char SATELLITES_QNT[10];
    char SAT_NAMES[30];
};

using namespace std;
int main() {
    const int N = 5;
    setlocale(LC_ALL, "Russian");
    PLANET SAT_NAMES[5];

    for (int i = 0; i < N; i++) {
        printf("Введите название планеты %i\n", (i + 1));
        scanf("%s", &SAT_NAMES[i].NAME);
        printf("Введите радиус планеты %i\n", (i + 1));
        scanf("%i", &SAT_NAMES[i].RADIUS_PL);
        printf("Введите радиус орбиты %i\n", (i + 1));
        scanf("%i", &SAT_NAMES[i].RADIUS_ORB);
        printf("Введите количество спутников %i\n", (i + 1));
        scanf("%s", &SAT_NAMES[i].SATELLITES_QNT);
    }
    for(int i = 0; i < N-1; i++)
    {
        for(int j = i + 1 ; j < N; j++)
        {
                                if(strcmp(SAT_NAMES[i].SATELLITES_QNT,
SAT_NAMES[j].SATELLITES_QNT)==0)
                                {
                                    printf(" Название планеты: %s\n", SAT_NAMES[ i ].NAME);
                                    printf(" Радиус планеты: %i\n", SAT_NAMES[ i ].RADIUS_PL);
                                    printf(" Радиус орбиты: %i\n", SAT_NAMES[ i ].RADIUS_ORB);
                                    printf("          Количество          спутников:          %s\n",
SAT_NAMES[ i ].SATELLITES_QNT);
                                }
        }
    }
}

```

```

    }
}
int A, B, t=0;
printf("Введите диапазон. От \n");
scanf("%i", &A);
printf("До \n");
scanf("%i", &B);
for(int i = 0; i <= N-1; i++)
{
    if((A <=
SAT_NAMES[ i ].RADIUS_ORB)&&(SAT_NAMES[ i ].RADIUS_ORB <= B))
    {
        t++;
        printf(" Название планеты: %s\n", SAT_NAMES[ i ].NAME);
        printf(" Радиус планеты: %i\n", SAT_NAMES[ i ].RADIUS_PL);
        printf(" Радиус орбиты: %i\n", SAT_NAMES[ i ].RADIUS_ORB);
        printf("      Количество      спутников:      %s\n",
SAT_NAMES[ i ].SATELLITES_QNT);
    }
}

if(t==0){
for (int i = 0; i < N; ++i)
{
    printf(" Название планеты: %s\n", SAT_NAMES[ i ].NAME);
    printf(" Радиус планеты: %i\n", SAT_NAMES[ i ].RADIUS_PL);
    printf(" Радиус орбиты: %i\n", SAT_NAMES[ i ].RADIUS_ORB);
    printf("      Количество      спутников:      %s\n",
SAT_NAMES[ i ].SATELLITES_QNT);
}

}

return 0;
}

```

Пример выполнения программы

```
Введите название планеты 1
q
Введите радиус планеты 1
1
Введите радиус орбиты 1
1
Введите количество спутников 1
1
Введите название планеты 2
e
Введите радиус планеты 2
1
Введите радиус орбиты 2
1
Введите количество спутников 2
1
Введите название планеты 3
r
Введите радиус планеты 3
1
Введите радиус орбиты 3
1
Введите количество спутников 3
1
Введите название планеты 4
t
Введите радиус планеты 4
3
Введите радиус орбиты 4
3
```

```
Радиус орбиты: 1
Количество спутников: 1
Название планеты: q
Радиус планеты: 1
Радиус орбиты: 1
Количество спутников: 1
Название планеты: e
Радиус планеты: 1
Радиус орбиты: 1
Количество спутников: 1
```

```
Введите диапазон. От
1
До
2
```

```
Название планеты: q
Радиус планеты: 1
Радиус орбиты: 1
Количество спутников: 1
Название планеты: e
Радиус планеты: 1
Радиус орбиты: 1
Количество спутников: 1
Название планеты: r
Радиус планеты: 1
Радиус орбиты: 1
Количество спутников: 1
```

```
Process returned 0 (0x0) execution time : 23.060 s
Press any key to continue.
```

Вывод изучили принципы описания строки и работы ними в СИ, функции для работы со строками, а также организацию ввода-вывода с использованием файлов.