

Министерство образования и науки
Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
«Самарский государственный аэрокосмический университет
имени академика С.П.Королёва
(национальный исследовательский университет)» (СГАУ)

**Работа с текстовыми файлами в программе
(Язык Си)**

Электронные методические указания к лабораторным
работам

Работа выполнена по мероприятию блока 1 «Совершенствование образовательной деятельности» Программы развития СГАУ на 2009 – 2018 годы по проекту «Разработка системы электронного дистанционного образования НИУ СГАУ по информатике, математике и физике»

Соглашение № 1/17 от .03.

Самара 2013

УДК004(075)

ББК32.9я7

P131

Составитель Михайлов Владимир Гаврилович

Рецензент: Семенов В.В., к.т.н., доцент кафедры математические методы в экономике

Работа с текстовыми файлами в программе (язык Си)

[Электронный ресурс] : электрон. метод. указания к лаб. работам / М-во образования и науки РФ, Самар. гос. аэрокосм. ун-т им. С. П. Королева (нац. исслед. ун-т); сост. В. Г. Михайлов. Электрон. текстовые и граф. дан. (0.12 Мбайт). - Самара, 2013. - 1 эл.опт. диск (CD-ROM).

В пособии приведены методические указания к лабораторной работе по дисциплине « Информационные технологии».

Методические указания предназначены для студентов 5 факультета, обучающихся по направлению подготовки 210400.62 «Радиотехника» в 1, 2 семестрах.

Самарский государственный
аэрокосмический университет, 2013

Введение

В современных языках программирования процедуры ввода-вывода должны быть универсальными в том смысле, что они должны обеспечивать взаимодействие программы не только со штатными устройствами, такими как клавиатура и монитор, но и с другими внешними устройствами. В первую очередь – с файлами на диске или на других накопителях информации, что позволяет, во-первых, организовать обмен данными между разными программами и, таким образом, автоматизировать и разделить процессы сбора и обработки данных. Во-вторых, заранее подготавливать данные на любом стороннем компьютере для их последующей обработки.

Такие файлы данных могут иметь различную структуру в том числе – структуру текстовых файлов в стандарте MS-DOS. Формат таких файлов по сути повторяет формат организации экрана монитора в текстовом режиме, что позволяет использовать те же механизмы форматного ввода – вывода, которые используются в стандартных процедурах **printf** и **scanf**.

Текстовые файлы.

Текстовые файлы – это файлы, в которых информация представлена в текстовом виде посредством символов из набора ASCII. При этом каждый символ жестко связан с конкретной строкой и с позицией внутри строки (как на экране монитора). Никакая дополнительная информации, например, о размере или формате символов, в файле не присутствует.

ASCII – американский стандартный код для передачи информации (American Standard Code for Information Interchange).

По этому стандарту каждому символу соответствует свой числовой код (номер). Стандарт ASCII был разработан для телеграфа и в начальном варианте содержал только 128 символов, которые кодировались 7-ми разрядными двоичными числами (0 – 127).

Рабочие символы этого стандарта включали буквы только латинского алфавита, цифры и другие знаки с клавиатуры пишущих машинок (телеграфных аппаратов). Затем этот стандарт был расширен до 256 символов, а их код стал 8-ми разрядным (0 – 255). При этом, первые 128 символов никогда не меняются сохраняя исходный стандарт, а следующие 128 кодов (128 – 255) отводятся под буквы национальных алфавитов и различные нестандартные символы (в том числе, под элементы т.н. псевдографики).

Назначение символов:

| | |
|-----------|--|
| 0-31 | -управляющие коды; |
| 32–126 | -коды рабочих символов; |
| 127 | -служебный код; |
| 128 - 255 | - символы национальных алфавитов и элементы псевдографики. |

Важнейшие из управляющих кодов:

| Дестичный код | Обозначение | Действие |
|---------------|-------------|----------------------------------|
| 0 | NUL | нулевой оператор (без операции) |
| 7 | BEL | звуковой сигнал (звонок) |
| 8 | BS | пробел |
| 9 | HT | горизонтальная табуляция |
| 10 | LF | перевод строки |
| 11 | VT | вертикальная табуляция |
| 12 | FF | перевод страницы (подача бланка) |
| 13 | CR | возврат каретки (перевод строки) |
| 27 | ESC | переключение (переход) |

Информация в файле может разделяться на строки произвольной длины. Признаком конца строки служат два специальных символа с кодами 10 и 13. В конце файла присутствует символ с кодом 26.

Работа с файлами.

Обмен данными между программой и текстовым файлом осуществляют операторы **fscanf** и **fprintf**, которые работают как обычные операторы **scanf** и **printf**, но кроме своих обычных параметров должны содержать указатель на имя этого текстового файла.

Поэтому, прежде чем начинать работу с файлом, содержащим данные, этот файл должен быть 'открыт' с помощью функции **fopen** из стандартной библиотеки **stdio.h**. Для этого объявляется т.н. файловая переменная **F**, которая станет указателем на имя файла:

FILE *F;

Здесь ключевое слово **FILE** определяет файловый тип переменной **F**, после чего этой переменной можно присвоить параметры текстового файла:

F=fopen(<имя файла>,<режим обмена>);

Первым аргументом функции **fopen** является <имя файла>, которое оформляется в виде символьной строки, содержащей полное имя данного тестового файла. Вторым аргумент <режим обмена> также является символьной строкой, которая указывает, как этот файл будет использоваться. Возможными режимами являются: чтение ("**r**"), запись ("**w**") и добавление данных в конец файла ("**a**").

Если делается попытка открыть несуществующий файл для записи или добавления, то такой файл будет создан (если это возможно).

Открытие существующего файла на запись приводит к удалению его старого содержимого.

Попытка чтения несуществующего файла является ошибкой. Ошибки могут быть обусловлены и другими причинами (например, попыткой чтения из файла, не имея на то разрешения). При наличии какой-либо

ошибки функция возвращает нулевое значение указателя **NULL** (который для удобства также определяется в файле **STDIO.H**).

После того как файл ‘открыт’ можно выполнять операции записи или чтения данных командами:

```
fprintf(F,<аргументы>);  
fscanf(F,<аргументы>);
```

Первым параметром в этих функциях должна указываться файловая переменная **F**

Текстовые файлы относятся к типу последовательных файлов. Очередные данные записываются или читаются только поочерёдно от начала к концу файла. Прочитать какую либо запись без прочтения всех предыдущих записей невозможно.

Достижение конца файла подтверждается логической функцией **feof(F)**, возвращающей в этом случае значение ‘истина’.

После завершения работы с файлом связь с ним прекращается (файл ‘закрывается’) командой **fclose(F)**. и файловая переменная **F** может использоваться для организации связи с другим файлом.

Ниже приведён пример записи числовых значений в текстовый файл **PRIM.TXT** и чтения этих значений из файла с последующим их отображением на экране монитора:

```
#include <stdio.h>  
#include <conio.h>  
  
main()  
{  
int a,b,c;  
FILE *f;  
clrscr();
```

```

printf("Ввод переменных A, B, C с
клавиатуры\n\n");          // блок 1
scanf("%d %d %d", &a,&b,&c);

f=fopen("prim.txt","w");
// блок 2 fprintf(f,"%d %d
%d\n",a,b,c); fclose(f);

f=fopen("prim.txt","r");
// блк 3 fscanf(f,"%d %d %d",
&a,&b,&c); printf("\nПечать переменных A,
B, C из файла\n\n");
printf("%d %d %d\n",a,b,c);
fclose(f);
}

```

В данном примере в блоке 1 для переменных **a,b,c** с клавиатуры вводятся числовые целые значения (рис.1).

В блоке 2 открывается файл PRIM.TXT для записи (режим “**w**” в аргументах функции **fopen**).

Указатель устанавливается в начало файла и по команде **fprintf** в файл заносится строка данных с указанным форматом (рис.2).

После записи файл PRIM.TXT закрывается.

В блоке 3 файл PRIM.TXT открывается для чтения данных (режим “**r**”). Указатель устанавливается в начало файла и по команде **fscanf** производится форматное считывание данных для переменных **a,b,c**, которые затем выводятся на экран монитора (рис.1).

Ввод переменных А, Б, С с клавиатуры

Печать переменных А, Б, С из файла

Рис. 1 Изображение экрана монитора

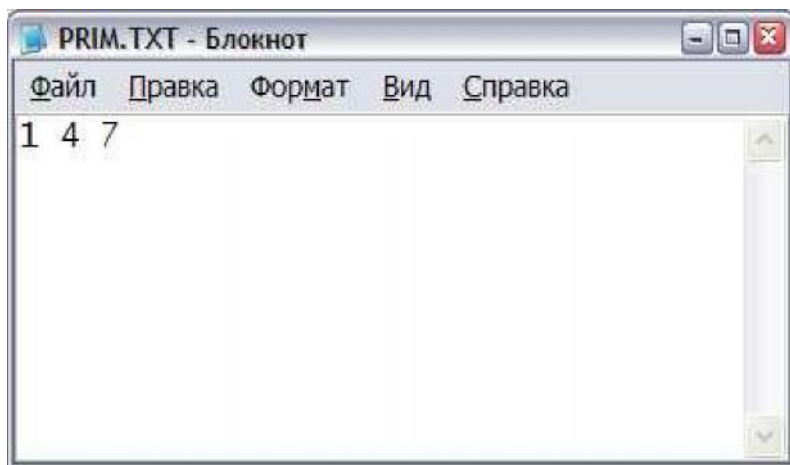


Рис.2 Содержимое текстового файла PRIM.TXT

Задание к лабораторной работе.

Вариант 1.

Создать текстовый файл в формате MS-DOS (одним из механизмов его создания может стать непосредственно редактор Турбо СИ).

Средствами редактора в файл поместить построчно пары целочисленных координат элементов **X** и **Y**, образующих изображение некоторой фигуры в центре экрана монитора в текстовом режиме. Варианты таких изображений необходимо согласовать с преподавателем.

Составить программу, которая по этим координатам восстановит соответствующее изображение в центре экрана монитора. Изображение может собираться как из элементов псевдографики, так и с помощью функции **window**.

Ниже приводится возможный вариант подобной программы, предусматривающий предварительное занесение данных в файл средствами какого-либо текстового редактора. Прочитанные из файла данные предварительно помещаются в соответствующие массивы **X** и **Y**.

Элементы изображения формируются как отдельные окна размером в одно знакоместо.

Эта программа выводит изображение в левый верхний угол экрана. Необходимые рисунки для отчёта представлены ниже.

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

main()
{
int i,n, X[20], Y[20];

FILE *f;
textbackground(1);
clrscr();
```

```

    f=fopen("prim 2.txt","r");           //
блок 1
    for (i=0; !feof(f); i++)
        fscanf(f,"%d %d", &X[i],&Y[i]);
    n=i-1;

    for (i=0; i<n; i++)
        //блок 2
        {
            window(X[i],Y[i],X[i],Y[i]);
            textbackground(7);
            clrscr();
        }
    fclose(f); }

```

На рис. 3 представлено изображение фигуры, которую требуется отобразить на экране.

На рисунке 4 показано содержимое текстового файла PRIM_2.TXT, в который были занесены пары координат X и Y элементов будущего рисунка (изображение находится в левом верхнем углу экрана).

В тексте программы блок 1 операторов открывает файл PRIM_2.TXT и построчно считывает пары координат. Для сохранения координат в данном примере используются два массива целого типа X и Y.

Чтение данных выполняется в цикле for до тех пор, пока не достигнут конец файла (**!feof(f)**).

В блоке 2 по этим координатам формируется изображение на экране. В данном случае элементы изображения выводятся на экран командой window как отдельные окна размером в одно знакоместо (рис. 3).



Рис.3 Изображение экрана с заданным рисунком.

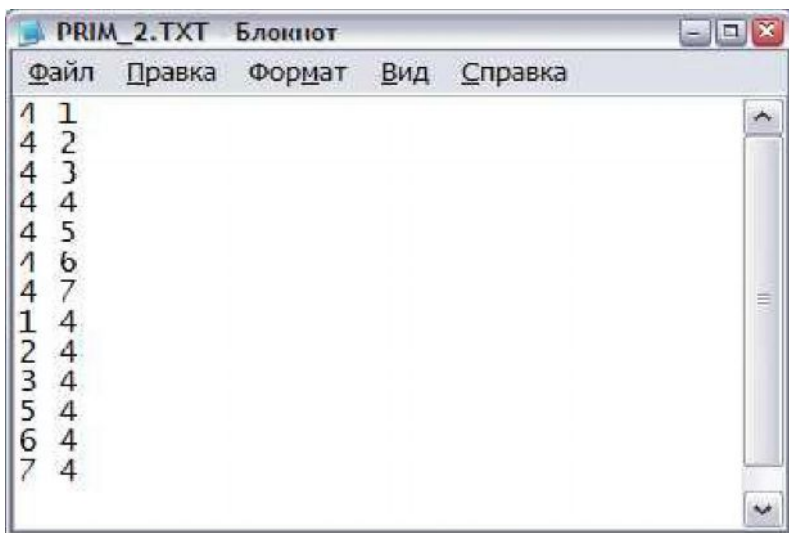


Рис. 4 Содержимое текстового файла PRIM_2.TXT с координатами элементов отображаемой фигуры.

Вариант 2.

Формирования текстового файла и занесение в него данных производится в одной программе с процедурой чтения данных из файла:

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

main()
{
    int i,n,x,y, XX[10], YY[10];

    FILE *f;
    textbackground(1);
    clrscr();

    f=fopen("prim 3.txt","w" ) ;           //
блок 1
    for (i=0; i<10; i++){
        printf(",Введи x, y\n");
        scanf("%d %d", &x,&y);
        fprintf(f,"%d %d\n",x,y);
    }
    fclose (f);

    f=fopen("prim 3.txt","r");
    // блок 2
    for (i=0; !feof(f); i++)
        fscanf(f,"%d %d", &XX[i],&YY[i]);
    clrscr(); for (i=0; i<10; i++)
    {
```

```
        window(XX[i],YY[i],XX[i],YY[i]) ;  
        textbackground(7); clrscr(); }  
fclose (f);  
}
```

В блоке 1 открывается текстовый файл PRIM_3.TXT в режиме записи данных, в который затем с клавиатуры вводятся пары координат элементов будущего изображения (рис. 6).

Получившийся список координат показан на рисунке 5.

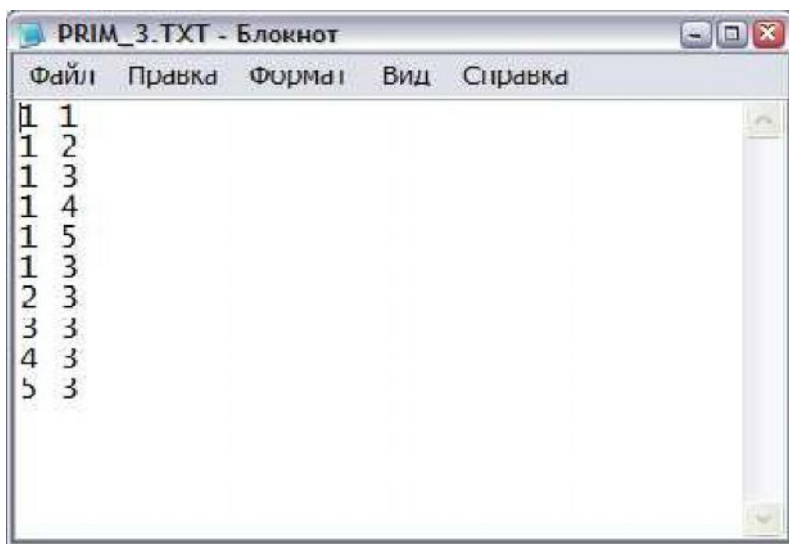


Рис. 5 Содержимое текстового файла PRIM_3.TXT.

Поле завершения ввода данных этот файл закрывается.

В блоке 2 файл вновь открывается в режиме чтения с той же файловой переменной **F**.

Из файла координаты элементов считываются и переносятся в соответствующие массивы **XX** и **YY**.

По этим координатам на экран монитора выводится заданное изображение (рис. 6):

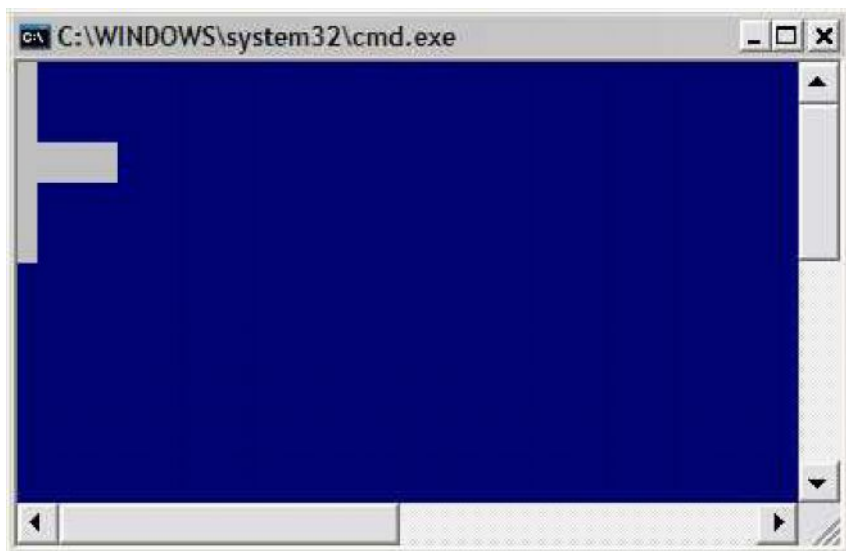


Рис. 6. Изображение на экране монитора, сформированное по координатам из файла PRIM_3.TXT.

Учебное издание

**Работа с текстовыми файлами в программе.
(Язык Си)**

Электронные методические указания к лабораторной работе

Составители: Михайлов Владимир Гаврилович

Редактор

Корректор

Лицензия

Подписано в печать . Формат 60x84 1/16.

Бумага офсетная. Печать офсетная.

Усл. печ. л. . Усл. кр.-отт. . Уч.-изд. л. .

Тираж 100 экз. Заказ . Арт. /2013

Самарский государственный аэрокосмический университет имени академика С.П. Королева 443086 Самара, Московское шоссе, 34

РИО Самарского государственного аэрокосмического университета.
443001, Самара, Московское шоссе, 34.

