**Семинар по текстовым файлам**

Файл – последовательность информационных записей. Такая последовательность может быть записана на дисках внешней памяти (магнитных, оптических и т.д.) или появляться в результате обмена информацией с внешними устройствами (клавиатура, устройство вывода на печать, дисплей и т.д.). Соответственно различают дисковые файлы и логические устройства.

Логически файлы представляют собой длинную строку байтов, которая разбивается на части (записи) в зависимости от типа файла. В С++ определены два типа файлов: **текстовые и двоичные**.

**Текстовые файлы** состоят из символьных строк переменной длины, заканчивающихся маркером «Конец строки» (двухбайтовый код: 13, 10 0D0A).

**Двоичные файлы** представляют собой последовательность компонентов фиксированного размера, например, целых чисел, структур и т. д., записанных в файл без преобразования к символьному виду.

Программирование операций ввода/вывода с файлами начинается с объявления файловой переменной:

**FILE \*<Файловая переменная>;** // объявляется указатель на таблицу FILE

Перед работой файл должен быть открыт. При открытии файла выделяется память под таблицу, на которую указывает файловая переменная, и частично заполняются ее поля.

Операцию выполняет специальная функция:

**<Файловая переменная>=fopen(<Имя файла>,<Операция [+] [Тип]>);**

Где:

**<Имя файла>** – строковая переменная или константа (это либо полное имя файла с путем, либо имя файла без пути. В этом случае файл ищется только в текущем каталоге;

**<Операция[+]>** – кодируется следующим образом:

r – ввод из существующего файла;

w – вывод с очисткой файла или создание нового файла для вывода;

a – добавление к существующему файлу или создание файла для вывода;

r+ – ввод/вывод в существующий файл;

w+ – ввод/вывод в существующий или создание нового файла для ввода/вывода;

a+ – ввод/добавление к существующему или создание файла для ввода/вывода.

**<тип>** – тип файла:

t – текстовый файл (принимается по умолчанию);

b – двоичный файл. **FILE \*f;**

**а) f=fopen(″abc.txt″,″w″);**

**б) f=fopen(″f:\\iva\\text.txt″,″r″);**

**1. ТЕКСТОВЫЕ ФАЙЛЫ**

**Текстовым называется файл**, состоящий из символьных строк переменной длины. При записи в файл, вся числовая информация преобразуется из внутреннего представления в символьное, и наоборот. При чтении числовой информации, она из символьного вида переводится во внутреннее представление. Поэтому, текстовые файлы иногда называют **файлами с преобразованием.**

Для работы с текстовыми файлами, содержащими только символьную информацию, используются функции ввода\вывода символов и строк:

**Ввод символа:**

**int getc(FILE \*stream);** // введенный символ возвращается как результат функции. При достижении конца файла функция возвращает константу **EOF**.

**Вывод символа:**

**int putc(int c,FILE \*stream);** // выводимый символ передается через параметр c, он же возвращается в качестве результата функции .

**Ввод строки:**

**char \*fgets(char \*s,int n,FILE \*stream);** // вводит строку длиной до n-1 или до маркера «Конец строки» в буфер по адресу s. Возвращает дубликат указателя на введенную строку. При достижении конца файла возвращает NULL.

**Вывод строки:**

**int fputs(const char \*s,FILE \*stream);** // выводит строку из буфера по адресу s. Если вывод прошел нормально, то возвращает 0, иначе возвращает -1.

Для работы с файлами, содержащими числовую или смешанную информацию, используются функции форматного ввода\вывода.

**Форматный ввод:**

**int fscanf(FILE \*stream,const char \*format[,adress,...]);** // Функция работает аналогично scanf (и scanf\_s), но при этом осуществляется ввод не с клавиатуры, а из указанного файла. Возвращает количество введенных полей.

Если при выполнении операции достигается конец файла, то функция возвращает **EOF**.

**Форматный вывод:**

**int fprintf(FILE \*stream,const char \*format[,argument,...]);** // Функция работает аналогично printf, но при этом вывод осуществляется в указанный файл.

**Пример 1.** **Сформировать текстовый файл строк, каждая из которых состоит из случайного числа одинаковых символов латинского алфавита. Вывести полученный файл на экран.**

**#include "stdafx.h"**

**#include <locale.h>**

**#include <stdio.h>**

**#include <conio.h>**

**#include <stdlib.h>**

**#include <time.h>**

**int main(int argc,char \*argv[ ])**

**{ srand((unsigned)time(NULL));**

**setlocale(0,"russian");**

**FILE \*f;**

**int n,k=26;**

**char s[80];**

**char b='A';**

**f=fopen("test.dat","w+");**

**for(int i=0;i<k;i++)**

**{ n= rand()%20+1;**

**for( int j=0;j<n;j++)**

**{ s[j]=b;**

**s[n]='\0';**

**}**

**b++;**

**fputs(s,f); // запись в файл**

**fputs("\n",f);**

**}**

**rewind(f);// Установка указателя файла на начало**

**while (fgets(s,80,f)!=NULL)**

**printf("%s",s);**

**fclose(f);**

**puts("Нажмите любую клавишу для завершения...");**

**\_getch();**

**return 0;**

**}**

**Результат работы**

****

**Программа Filstrput**

**Пример 2. Сформировать текстовый файл программным путем из вводимых строк переменной длины. Вывести файл на экран. В сформированном файле найти строки, длина которых меньше указанной, вводимой с клавиатуры.**

**#include "stdafx.h"**

**#include <locale.h>**

**#include <stdio.h>**

**#include <conio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**int main(int argc,char \*argv[ ])**

**{ setlocale(0,"russian");**

**FILE \*f;**

**int n,k=0;**

**char s[80];**

**puts("Input string or Enter");**

**f=fopen("test.dat","w+");**

**while(gets(s),strlen(s)!=0*)***//Цикл формирования файла

***{ fputs(s,f);***

***fputs("\n",f);***

**}**

**rewind(f);**//Установка указателя файла на начало

**puts("Inputed File");**

**while (fgets(s,80,f)!=NULL)**

//Печат строки и ее длины с учетом сивола \n **0D0A**

**printf("%4d %20s",strlen(s),s);**

**rewind(f);**

**puts("Input length for find");**

**scanf("%d",&n);**

**while (fgets(s,80,f)!=NULL)**

**if(strlen(s)<(n+1))**//длина строки с учетом **0D0A**

**{**

**printf("%s",s);**

**k++;**

**}**

**printf("In File %5d string with length <%5d\n",k,n);**

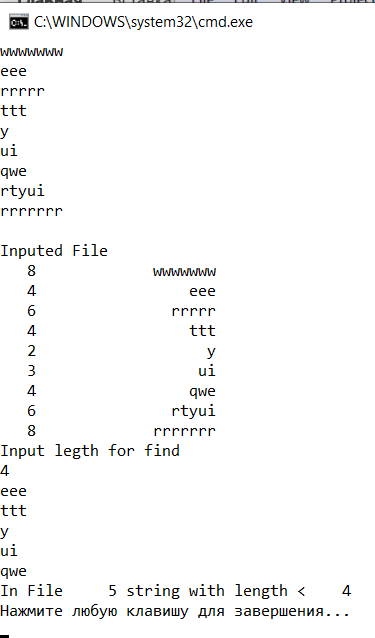
**fclose(f);**

**puts("Нажмите любую клавишу для завершения...");**

**\_getch();**

**return 0;**

**}**



**Программа Filstrlen**

**Пример 3. Дан текстовый файл строк, состоящих из слов через пробел. Переписать в другой файл все строки, состоящие из одного слова. В новом файле определить слово максимальной длины и его номер в файле**

**#include "stdafx.h"**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**#include <time.h>**

**#include <conio.h>**

**void printfile(FILE \*f)//процедура печати файла**

**{ rewind(f);**

**char s[80];**

**while(fgets(s,80,f)!=NULL)**

**printf("%s",s);**

**}**

**//функция определения строки с одним словом**

**int isone(const char \*s)**

**{int i=0;**

**int f=0,fl=0;**

**while((i<=strlen(s))&&(fl==0))**

**{ if((s[i]==' ')||(s[i]=='\0'))**

**f++;**

**i++;**

**}**

**if (f>1)fl=1;**

**return fl;**

**}**

**//процедура определения максимального слова,длины, //номера**

**void maxsl(FILE \*f,char \* maxsl,int &maxdl,int &nummax)**

**{ char s[20];maxdl=0;maxsl[0]='\0';**

**nummax=0;int k=0;**

**rewind(f);**

**while(fgets(s,20,f)!=NULL)**

**{ int l=strlen(s)-1; //поправка на 0D0A**

**k++;**

**if(l>maxdl)**

**{ strcpy(maxsl,s);**

**maxdl=l;**

**nummax=k;**

**}**

**}**

**}**

**int main(int argc, char\* argv[])**

**{ //int i,r;**

**FILE \*f1,\*f2;**

**char s[80],slovo[20];**

**int dl,num=0;**

**f1=fopen("F1.dat","r");**

**f2=fopen("F2.dat","w+");**

**puts("\_\_\_\_\_FILE 1\_\_\_\_\_\_\_\_\_");**

**printfile(f1);**

**rewind(f1);**

**while (fgets(s,80,f1)!=NULL)**

**{ if(isone(s)==0)**

**fputs(s,f2); //так как 0D0A в строке есть**

**}**

**puts("\_\_\_\_\_FILE 2\_\_\_\_\_\_\_\_\_");**

**printfile(f2);**

**maxsl(f2,slovo,dl,num);**

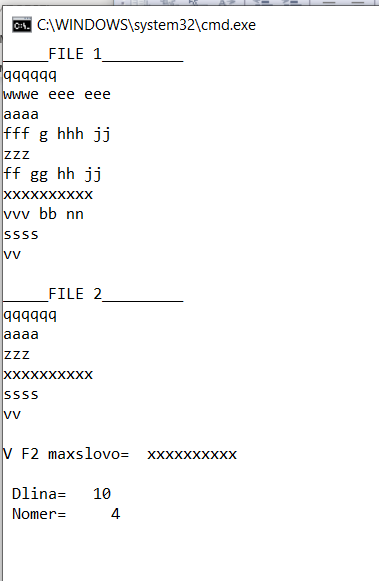
**printf("V F2 maxslovo= %s\n Dlina=%5d\n Nomer= %5d\n",slovo,dl,num);**

**fcloseall();**

**getch();**

**return 0;**

**}**

****

**Пример 4. Дан текстовый файл строк, состоящий из слов через пробел. Переписать в другой файл строки, содержащие слово, введенное с клавиатуры**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**#include <time.h>**

**#include <conio.h>**

**void printfile(FILE \*f)**

**{ rewind(f);**

**char s[80];**

**while(fgets(s,80,f)!=NULL)**

**printf("%s",s);**

**}**

**int isslovo(const char \*s,char \*sl)**

**{char simtok[]=" \0";**

**char \*tok;int f=0;**

**char \*ss=new char[strlen(s)+1];**

**strcpy(ss,s);**

**tok=strtok(ss,simtok);**

**while((tok!=NULL)&&(f==0))**

**{ // puts(tok);**

**if(strcmp(sl,tok)==0)**

**f=1;**

**tok=strtok(NULL,simtok);**

**}**

**delete [] ss;**

**return f;**

**}**

**void main()**

**{ int i,r;**

**FILE \*f1,\*f2;**

**char s[80],slovo[15];**

**f1=fopen("F1.dat","r");**

**f2=fopen("F2.dat","w+");**

**puts("\_\_\_\_\_FILE 1\_\_\_\_\_\_\_\_\_");**

**printfile(f1);**

**rewind(f1);**

**puts("Input slovo for find");**

**gets(slovo);**

**while (fgets(s,80,f1)!=NULL)**

**{ if(isslovo(s,slovo)==1)**

**fputs(s,f2);**

**}**

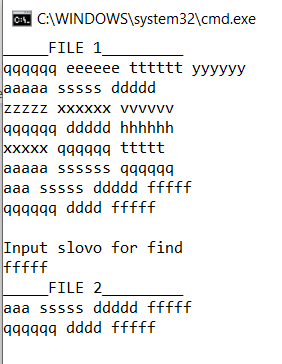
**puts("\_\_\_\_\_FILE 2\_\_\_\_\_\_\_\_\_");**

**printfile(f2);**

**fcloseall();**

**getch();**

**}**

****

**F1F1SlovoDel2**

**ФОРМАТНЫЙ ВВОД\_ВЫВОД**

**Пример 5. Сформировать текстовый файл целых чисел. Переписать в другой файл все числа, кратные 2. Вывести на экран оба файла.**

**#include "stdafx.h"**

**#include <conio.h>**

**#include <stdio.h>**

**#include <string.h>**

**#include <time.h>**

**#include <locale>**

**int \_tmain(int argc, \_TCHAR\* argv[])**

**{**

**setlocale(0,"");**

**srand((unsigned)time(NULL));**

**int N,k,i;**

**FILE \*f,\*f1,\*d;**

**char name[30];**

**N=rand()%50+10;**

**printf("Количество вводимых элементов - %d\n",N);**

**puts("Введите имя файла");**

**scanf("%s",&name);**

**f=fopen(name,"w+");**

**d=f;**

**for(i=0;i<N;i++)**

**{**

**k=rand()%101-rand()%101;**

**fprintf(f,"%d ",k);**

**}**

**rewind(f);**

**puts("\_\_\_\_\_\_\_\_FILE1\_\_\_\_\_\_\_\_\_\_\_\_\_");**

**while(!feof(f))**

**{**

**fscanf(f,"%d ",&k);**

**printf("%d ",k);**

**}**

**printf("\n");**

**rewind(f);**

**puts("\nВведите название второго файла");**

**scanf("%s",&name);**

**f1=fopen(name,"w+");**

**while(!feof(f))**

**{**

**fscanf(f,"%d",&k);**

**if(k%2==0)**

**fprintf(f1,"%d ",k);**

**}**

**rewind(f1);**

**puts("\_\_\_\_\_\_\_\_FILE2\_\_\_\_\_\_\_\_\_\_\_\_\_");**

**while(!feof(f1))**

**{**

**fscanf(f1,"%d ",&k);**

**printf("%d ",k);**

**}**

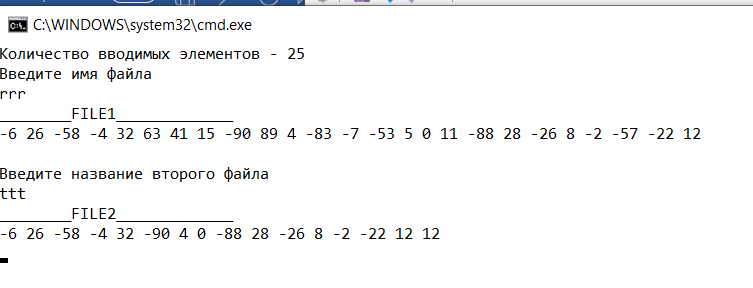
**printf("\n");**

**fcloseall();**

**getch();**

**return 0;**

**}**

****

**Программа РК**

**Пример 6. Дан текстовый файл, содержащий целые числа. Из чисел файла сформировать одномерный массив b[100], содержащий не более 100 элементов. Удалить из массива все отрицательные числа. Сформированный массив записать в другой текстовый файл с комментариями.**

**#include "stdafx.h"**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <time.h>**

**int b[100];**

**int i,j,n,e;**

**FILE \*f,\*r;**

**char s[15];**

**int main(int argc, char\* argv[])**

**{puts("File Name");**

**scanf("%s",s);**

**f=fopen(s,"r");**

**i=0;**

**while((i<100)&&(fscanf(f,"%d",&e)!=EOF))**

**{ b[i]=e;i++;**

**}**

**if(i<100)n=i;**

**else n=100;**

**puts("Inputed Massiv");**

**for(i=0;i<n;i++)**

**printf("%4d",b[i]);**

**printf("\n");**

**j=-1;**

**for(i=0;i<n;i++)**

**if (b[i]>=0)**

**{ j=j+1;**

**b[j]=b[i];**

**}**

**for(i=j+1;i<n;i++)**

**b[i]=0;**

**n=j+1;**

**if (n==0) puts("Empty Massiv n=0");**

**else**

**{**

**puts("File Name result");**

**scanf("%s",s);**

**r=fopen(s,"w");**

**fputs("New Massiv\n",r);**

**for(i=0;i<n;i++)fprintf(r,"%4d",b[i]);**

**fprintf(r,"\n");**

**fclose(r);**

**if((r=fopen(s,"r"))!=NULL)**

**{fgets(s,15,r);**

**puts(s);**

**while(fscanf(r,"%d",&e)!=EOF)**

**{**

**printf("%4d",e);**

**}**

**printf****("\n");**

**}**

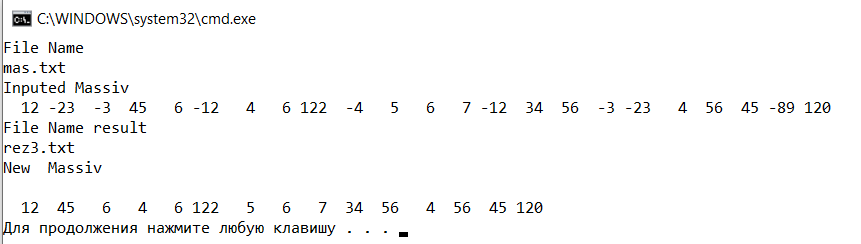
**else puts(("File not found\n");**

**}**

**fcloseall();**

**return 0;**

**}**



**Программа FileWW**

**Задание на дом**

**Задача 1.** Сформировать текстовый файл, состоящий из строк переменной длины, содержащей слова, записанные через один пробел.

Переписать из этого файла в новый те строки, количество слов в которых больше указанного.

**Задача 2.** Дан текстовый файл, содержащий вещественные числа. Сформировать их этого файла новый, все числа которого должны быть больше среднего арифметического чисел исходного файла.