

Министерство образования и науки Российской Федерации
Санкт-Петербургский политехнический университет Петра Великого

—
Институт компьютерных наук и технологий
Кафедра «Информационная безопасность компьютерных систем»

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2
« У д а л я т о р »
По дисциплине «Методы программирования»

Выполнили
Студент гр. 13508/13

А.Э.Палёный
А.Романов

Проверил
Преподаватель

В.Б.Вагисаров

Санкт-Петербург
2016

Цель

Создать программу, которая будет считывать текст и файла «Test.txt» и из файла «Del.txt». После этого программа должна удалить все слова, из первого файла, которые встречаются во втором. Результат должен быть записан в файл «Output.txt».

Ход работы

Была создана программа, удаляющая всю информацию из первого файла, которая была введена во второй и выводящая результат в третий файл.

Данная программа разбита на 2 основные части:

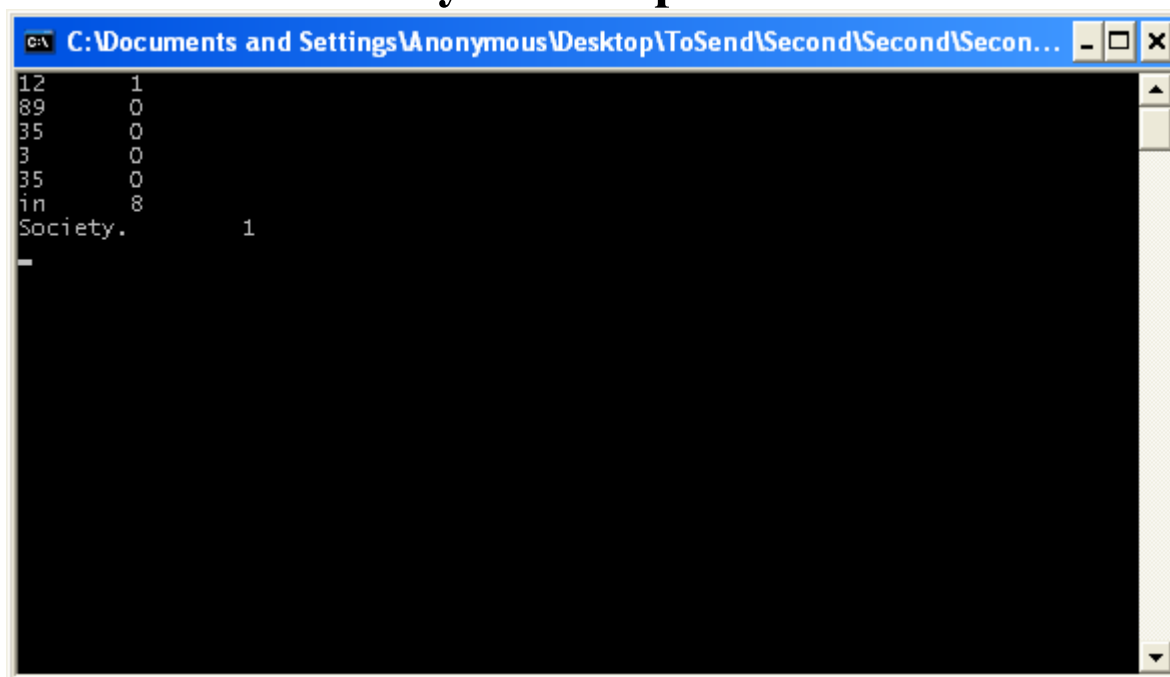
- 1) Тут происходит считывание слов из файла «Del.txt» и запись их в линейный список. Слова считываются до пробела.

После этого, этот линейный список передается во вторую часть.

- 2) В данной части происходит считывание слов посимвольно из файла «Test.txt» и проверка их на слова переданные в линейном списке из первой части.

Если слова совпадают, то происходит игнорирование, и слова не записываются в конечный файл «Output.txt», иначе – слова записываются и проверка продолжается до тех пор, пока в файле «Test.txt» не будет обнаружен конец (EOF).

Результаты работы



Файл «Del.txt»

12 89 89 35 3 35 in Society.

Файл «Test.txt»

As of 2015, the Society had published 344 volumes in its Original Series; 126 volumes in its Extra Series, published between 1867 and 1935, comprising texts previously printed, but only in unsatisfactory or rare editions; and 25 volumes in its Supplementary Series, an occasional and irregular series initiated in 1970. The Society keeps the majority of its older publications in print, excepting only those which have been superseded by subsequent editions in its own series. Volumes are currently published on behalf of the Society by Oxford University Press.[2]

Famous members of the society when it was formed in 1864 include Furnivall himself, Alfred Tennyson (poet laureate), Warren de la Rue (inventor of the lightbulb), Richard Chenevix Trench (Irish ecclesiastic), Stephen Austin (a Hertford-based printer), Edith Coleridge (granddaughter of Samuel Taylor Coleridge), the Rev. Richard Morris (the editor of 12 volumes between 1862 and 1880), and others.

The Society emblem is a modified representation of the Anglo-Saxon Alfred Jewel, incorporating a scroll bearing the name of the Society.

Файл «Output.txt»

As of 2015, the Society had published 344 volumes its Original Series; 126 volumes its Extra Series, published between 1867 and 1935, comprising texts previously printed, but only unsatisfactory or rare editions; and 25 volumes its Supplementary Series, an occasional and irregular series initiated 1970. The Society keeps the majority of its older publications print, excepting only those which have been superseded by subsequent editions its own series. Volumes are currently published on behalf of the Society by Oxford University Press.[2]

Famous members of the society when it was formed 1864 include Furnivall himself, Alfred Tennyson (poet laureate), Warren de la Rue (inventor of the lightbulb), Richard Chenevix Trench (Irish ecclesiastic), Stephen Austin (a Hertford-based printer), Edith Coleridge (granddaughter of Samuel Taylor Coleridge), the Rev. Richard Morris (the editor of volumes between 1862 and 1880), and others.

The Society emblem is a modified representation of the Anglo-Saxon Alfred Jewel, incorporating a scroll bearing the name of the

Листинг программы

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

struct List {
    char *word;//информационное поле
    List *next;//адресное поле
    int CountWord;
};

void Vivod(List **Top)
{
    List *Head;
    Head=*Top;
    *Top=Head->next;
}

void add(List **Top)
{
    struct List *Head = *Top;
    Head->next = new List();
    *Top=Head->next;
    Head->next->next=NULL;
}

void analyze(struct List **From, int *length)
{
    struct List *Top;
    struct List *Current, *Dump;

    FILE * Input;    //Файл для проверки слов
    char c,*p;
    int count =0;
    int times =0;
    int i,j;
    int    flag=1;

    Top = new List();
    p = new char;
    Dump=Current=Top;

    if( (Input = fopen("Del.txt","r")) != 0 )
    {
        while(fscanf(Input,"%c",&c)!=EOF)
        {
            count++;
            if((c==32 || c==10)&& count ==1){    c=0;    count=0;}
            if(c==32 || c==10)
            {
                p=(char*)realloc(p,count);
                *(p+count-1)=0;
                for(i=0; i<times; i++)
                {
                    if(*(Dump->word) == *p)
                    {
                        flag=0;
                        for(j=0; j!=count-1; j++)
                        {
                            if(*(Dump->word+j) != *(p+j)){    flag=1;

j=count-2;}

                        }
                        if(!flag && *(p+j)==0 && *(Dump->word+j)==0){    flag=0;

i=times-1;}

                        else flag=1;
                    }
                }
            }
        }
    }
}
```

```

        Vivod(&Dump);
    }
    Dump=Top;

    if(flag)
    {
        Current->word=p;
        add(&Current);
        times++;
        p=new char;
    }else p=(char*)realloc(p,1);
    count=0;
    flag=1;
    }else{ p=(char*)realloc(p,count); *(p+count-1)=c;}
}

p=(char*)realloc(p,count+1);
*(p+count)=0;

//Проверка на существование таких же слов
for(i=0; i<times; i++)
{
    if(*(Dump->word) == *p)
    {
        flag=0;
        for(j=0; j!=count; j++)
        {
            if(*(Dump->word+j) != *(p+j)){ flag=1; j=count-1;}
        }
        if(!flag && *(p+j)==0 && *(Dump->word+j)==0){ flag=0; i=times-1;}
        else flag=1;
    }
    Vivod(&Dump);
}
Dump=Top;

if(flag)
{
    Current->word = p;
    add(&Current);
    times++;
}else free(p);

*From=Top;
*length=times;
}else printf("The file doesn't exist");
}

void Check(List **Res, int *len)
{
    FILE * Input;    //Файл для проверки слов
    FILE * Output;   //Файл для перезаписи
    char c;
    char *p;
    int count=1;
    int i=1;
    int j;
    int a;
    int crp=0;
    short flag;
    struct List *From, *Dump;
    int length;

    analyze(&From, &length);
    Dump=From;
    *Res=From;
    *len=length;
}

```

```

p=new char;

if( (Input = fopen("Test.txt","r")) != 0 )
{
    Output = fopen("Output.txt","w");           //Создаем файл

    while(fscanf(Input,"%c",&c)!=EOF)
    {
        flag=0;
        p=(char*)realloc(p,count);
        *(p+count-1)=c;

        if(c==32 || c==10)
        {
            flag=1;
            p=(char*)realloc(p,count);
            *(p+count-1)=0;
            for(i=0; i<length; i++)
            {
                if(*(Dump->word) == *p)
                {
                    flag=0;
                    for(j=0; j!=count-1; j++)
                    {
                        if(*(Dump->word+j) != *(p+j)){          flag=1;

j=count-2;}

                    }
                    if(!flag && *(p+j)==0 && *(Dump->word+j)==0){
                        flag=0;  i=length-1;          p=(char*)realloc(p,1);

count=0;

                        Dump->CountWord++;}
                    else flag=1;

                }
                Vivod(&Dump);
            }
            Dump=From;
        }

        if(flag)
        {
            *(p+count-1)=c;
            fwrite(p, sizeof(char), count, Output);           //Записываем валидную строку

            count=0;
            p=(char*)realloc(p,1);
        }
        count++;
    }

//Проверка последнего слова

    flag=1;
    p=(char*)realloc(p,count);
    *(p+count-1)=0;

    for(i=0; i<length; i++)
    {
        if(*(Dump->word) == *p)
        {
            flag=0;
            for(j=0; j!=count-1; j++)
            {
                if(*(Dump->word+j) != *(p+j)){          flag=1;  j=count-2;}

            }
            if(!flag && *(p+j)==0 && *(Dump->word+j)==0){          flag=0;  i=length-1;

Dump->CountWord++;}

            else flag=1;
        }
    }
}

```

```

        }
        Vivod(&Dump);
    }

    if(flag)
    {
        fwrite(p, sizeof(char), count-1, Output);           //Записываем валидную
строку
    }
    free(p);
    //Освобождаем

    fclose(Input);
    fclose(Output);
    //Закрываем файл для записи

} else    printf("The file doesn't exist");

}

int main()
{
    struct List *From;
    int length,i;

    time_t t=clock();

    Check(&From, &length);

    for(i=0; i<length; i++)
    {
        printf("%s",From->word);
        printf("\t%d\n",From->Count Word);
        Vivod(&From);
    }
    printf("Time = %d",clock()-t);
    getchar();
    return 0;
}

```