

## 11.ВЫДЕЛЕНИЕ СЛОВ В ФАЙЛАХ

**Цель работы** – изучение способов выделения последовательности символов из строк файла для последующей обработки.

### 11.1 Подготовка к лабораторной работе

При подготовке к лабораторной работе необходимо изучить методы удаления или пропуска разделителей в строках; способы выделения, удаления, обработки и вставки слов.

### 11.2 Теоретические сведения

Часто при обработке строк требуется выделять слова. Словом будем считать последовательность любых символов, отличных от пробелов или символов-разделителей. Задача выделения слов является важной подзадачей в лексическом анализе текста программы. В этом случае слова называются лексемами и определяются как минимальные единицы языка, имеющие самостоятельный смысл.

Рассмотрим алгоритм пропуска разделителей и выделения слов в том случае, когда строка не должна меняться, а слова разделяются знаками препинания.

Цикл для пропуска разделителей между словами:

```
while ((isspace(str[i])||ispunct(str[i]))&& i < str.size()-1)
    ++i; //пропустить пробелы и любые разделители
```

Слова можно пропустить аналогичным циклом:

```
while ( !isspace(str[i])&&!ispunct(str[i])&& i < str.size()-1)
    ++i; // пропустить все символы слова
```

Эти два цикла должны быть включены во внешний цикл, который закончится тогда, когда закончится строка.

Пример вывода на экран всех слов строки str:

```
int i = 0; str+=" ";
while (i < str.size()-1)
{
    while ((isspace(str[i])||ispunct(str[i]))&& i < str.size()-1)
        ++i; //пропустить пробелы и любые разделители
    begin = i; // номер первого символа слова
    while ( !isspace(str[i])&&!ispunct(str[i])&& i < str.size()-1)
        ++i; // пропустить все символы слова
    end = i; // номер символа, следующего за последним символом слова
    s=str.substr(begin, end-begin) ;// копируем в s (end-begin)символов строки str, начиная с begin
    cout<<s<<"\n";//вывод слова
}
```

**Пример 11.1.** Дан файл с текстовой информацией. Получить новый файл, ограничив длину всех слов (слова разделены пробелами) до 6 символов, если слово короче, то добавить '!'.

```
#include <iostream>
#include <fstream> //для работы с файлами
#include <string>
#include <locale>
using namespace std;
string Neg(string st)
{
    string nst, s;
    st+=" ";
    int len = st.size() ; //длина введённого текста
    int i = 0;
```

```

int begin = 0, end = 0;
while (i < st.size()-1)
{
    while (st[i] == ' ' && i < st.size()-1)
        ++i;                                //пропустить пробелы
    begin = i;                                // номер первого символа слова
    while (st[i] != ' ' && i < st.size()-1)
        ++i;                                // пропустить символы слова

    end = i;    // номер символа, следующего за последним символом слова

    //сформировать новую строку, ограничив длину всех слов до 6, если слово короче, то
добавить !
    s=st.substr(begin, end-begin); // копируем в s (end-begin)символов строки str, начиная с begin
    s.resize(6, '!');
    nst+=s+' ';
}
cout<<nst<<"\n";
return nst;
}

int main()
{
    setlocale (LC_CTYPE,"rus");
    string str, nstr;
    ifstream fin;                //файл для чтения
    fin.open("e:\\x.txt");
    ofstream fout;               //файл для записи отсортированного массива
    fout.open("e:\\y.txt");
    if(!fin.is_open())           //если не удалось открыть файл
    {
        cout<<"Файл не найден";
        return 1;
    }
    while(!fin.eof())            //пока не конец файла
    {
        getline(fin,str,'\n'); //чтение строки из файла
        nstr=Neg(str);
        fout<< nstr <<"\n";     //запись строки в файл
    }
    fin.close();                //закрыть файл
    fout.close();
    return 0;
}

```

### 11.3 Варианты заданий

1. Поменять местами первое и последнее слово каждой строки. Слова разделяются пробелами.
2. Переставить в начало каждой строки все слова, состоящие из цифр.
3. Определить строку с максимальным количеством слов.
4. Переписать в новый файл те слова, которые содержат хотя бы одну цифру.
5. Найти в каждой строке те слова, в которых все символы упорядочены по возрастанию.
6. Заменить все слова, длина которых превышает 7, заданной строкой.
7. Во всех словах, которые состоят только из строчных латинских букв, заменить эти буквы прописными.
8. Среди всех слов файла найти слова, представляющие собой целые числа. Увеличить все эти числа на 1.
9. В каждой строке все слова с четными номерами переписать в новый файл.
10. Переписать в новый файл все слова, которые содержат только буквы латинского алфавита.
11. Удалить из файла те слова, которые начинаются и заканчиваются цифрами.
12. Удалить из файла те зарезервированные слова, которыми ограничивается составной оператор.
13. В каждой строке подсчитать количество служебных слов, которые используются для описания составных типов данных.

14. Переписать в новый файл те слова, которые являются идентификаторами.
15. В каждой строке текстового файла удалить те слова, которые являются знаками логических операций.
16. Переписать в новый файл те слова, длина которых четная.
17. Удалить из каждой строки односимвольные слова.
18. В каждой строке поменять местами слова с четными и нечетными номерами.
19. Переписать в новый файл те слова, которые содержат заданную последовательность символов.
20. Переписать в новый файл все строки исходного файла, заменив порядок следования слов на обратный.
21. Переписать в новый файл те слова, длина которых кратна заданному числу.
22. В каждой строке файла подсчитать количество слов, которые являются записью вещественного числа.
23. Определить в каждой строке те слова, в которых есть повторение первой буквы слова.
24. В словах с четной длиной слова заменить все символы, на символы, коды которых на 1 больше, то есть на следующие по порядку символы.
25. Определить, есть ли в файле строки, в которых повторяется первое слово строки.

#### **11.4 Контрольные вопросы**

1. Дайте определение понятию "лексема".
2. Как определить начало и конец слова, не удаляя лишних пробелов в строке?
3. Как проверить достижение конца файла?
4. Как прочитать строку из файла?
5. Изменяется ли строка при копировании из неё слова в новую строку?