Федеральное агентство связи Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

Кафедра прикладной математики и кибернетики

**Курсовая работа по дисциплине**

**«Программирование»**

Выполнил: студент I курса ИВТ,

гр. ИП-811

Мироненко К.А.

Проверил преподаватель

Перцев И. В.

Новосибирск 2019

# Оглавление

[***Введение*** 2](#_Toc9172523)

[***Описание курсового проекта*** 3](#_Toc9172524)

[**Файл structure.hpp:** 3](#_Toc9172525)

[**Файл functions.hpp:** 3](#_Toc9172526)

[**Файл functions.сpp:** 4](#_Toc9172527)

[**Файл sort.hpp:** 6](#_Toc9172528)

[**Файл sort.сpp:** 7](#_Toc9172529)

[**Файл main.cpp:** 8](#_Toc9172530)

[***Пример работы программы*** 10](#_Toc9172531)

# Введение

**Вариант курсовой:** 11

**Задача курсовой работы:** «Считать из текстового файла слова в двусвязный список. Отсортировать символы каждого слова в алфавитном порядке. Сформировать новый список, таким образом чтобы слова были упорядочены в алфавитном порядке. Результат вывести в файл.»

**Используемые библиотеки:**

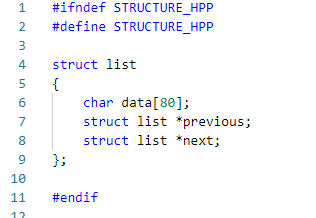
* iostream – для организации ввода-вывода
* string.h – для использования функции strcmp, используемой для сравнения массивов char

# Описание курсового проекта

Проект состоит из трех cpp-файлов: main.cpp, functions.cpp и sort.cpp, и трех заголовочных файлов: structure.hpp, sort.hpp, functions.hpp. При сборке из cpp-файлы компилируются в объектные файлы(.o) а из них собирается исполняемый файл (.exe)

## **Файл structure.hpp:**

Содержит описание используемой структуры

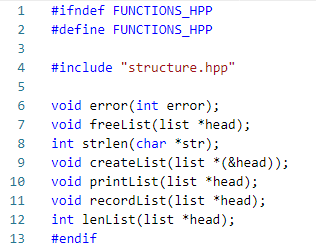


Объявляется структура list с полями:

* Data (тип char[])- используется для хранения слов
* previous (тип list \*) – указатель на предыдущую ячейку памяти, хранящую list
* next (тип list \*) – указатель на последующую ячейку памяти, хранящую list

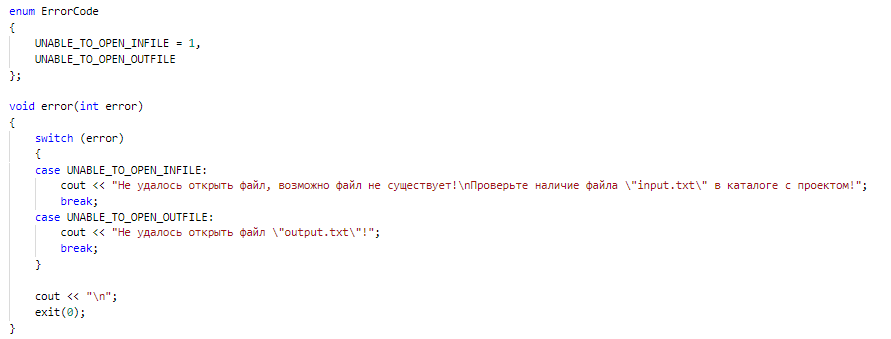
## **Файл functions.hpp:**

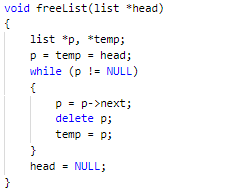
Содержит предобъявление основных функций, используемых в проекте



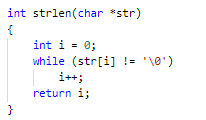
* void error(int error); - используется для обработки ошибок
* void freeList(list \*head); – используется для освобождения памяти
* int strlen(char \*str); - используется для определения длины строки
* void createList(list \*(&head)); - используется для создания двусвязного списка из файла
* void printList(list \*head); - используется для создания вывода списка
* void recordList(list \*head); - используется для записи измененного списка в файл
* int lenList(list \*head); - используется для определения длины списка

## **Файл functions.сpp:**

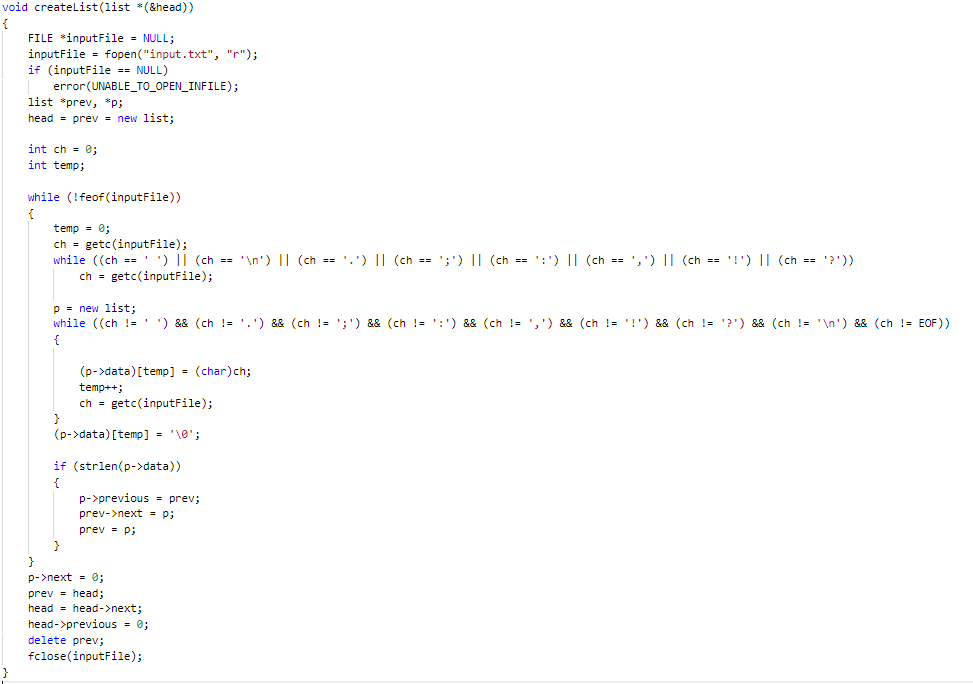
* void error(int error); 
* void freeList(list \*head);

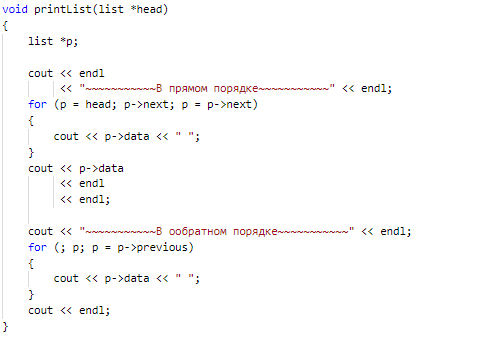


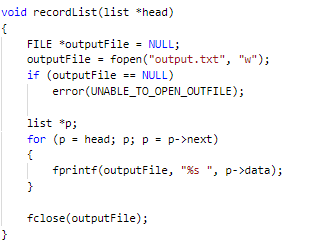
* int strlen(char \*str);



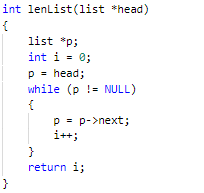
* void createList(list \*(&head));



* void printList(list \*head); 
* void recordList(list \*head);

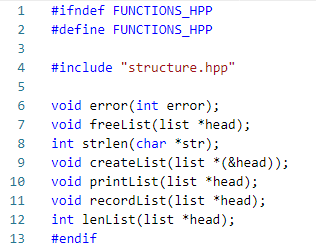


* int lenList(list \*head);



## **Файл sort.hpp:**

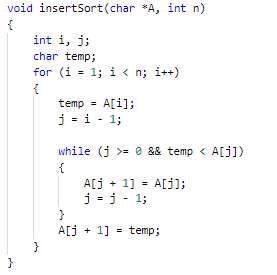
Содержит предобъявление функций, используемых для сортировки списка



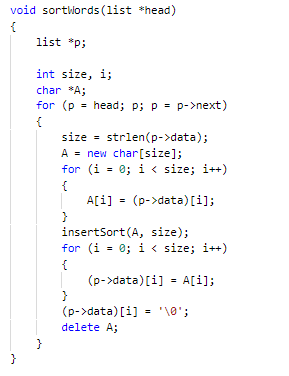
* void insertSort(char \*A, int n) - insert сортировка
* void sortWords(list \*head); - функция для сортировки слова, использует insertSort
* void sortList(list \*(&head)); - функция для сортировки списка

## **Файл sort.сpp:**

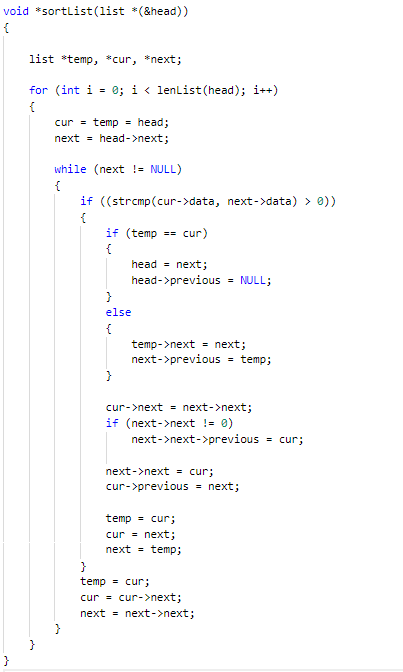
* void insertSort(char \*A, int n);



* void sortWords(list \*head);

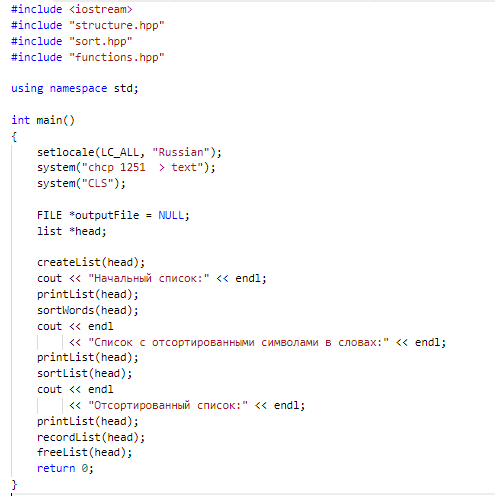


* void sortList(list \*(&head));

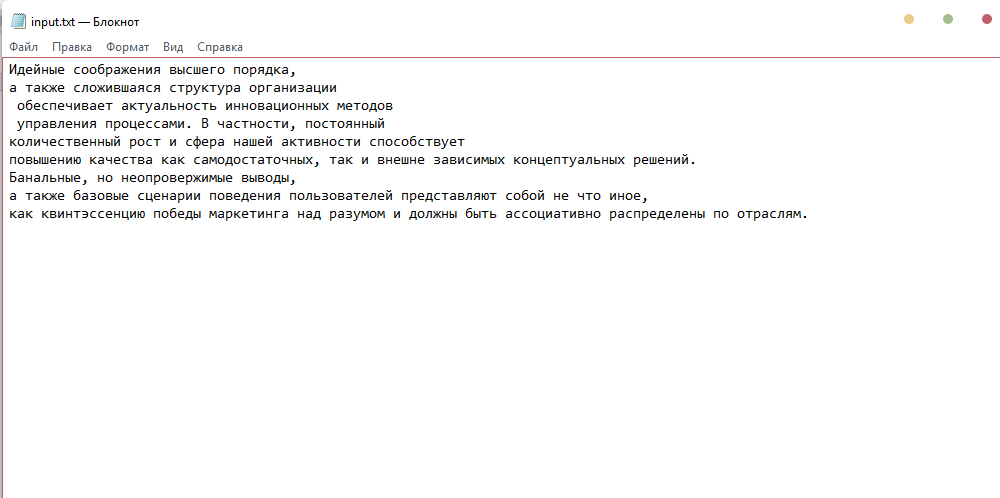


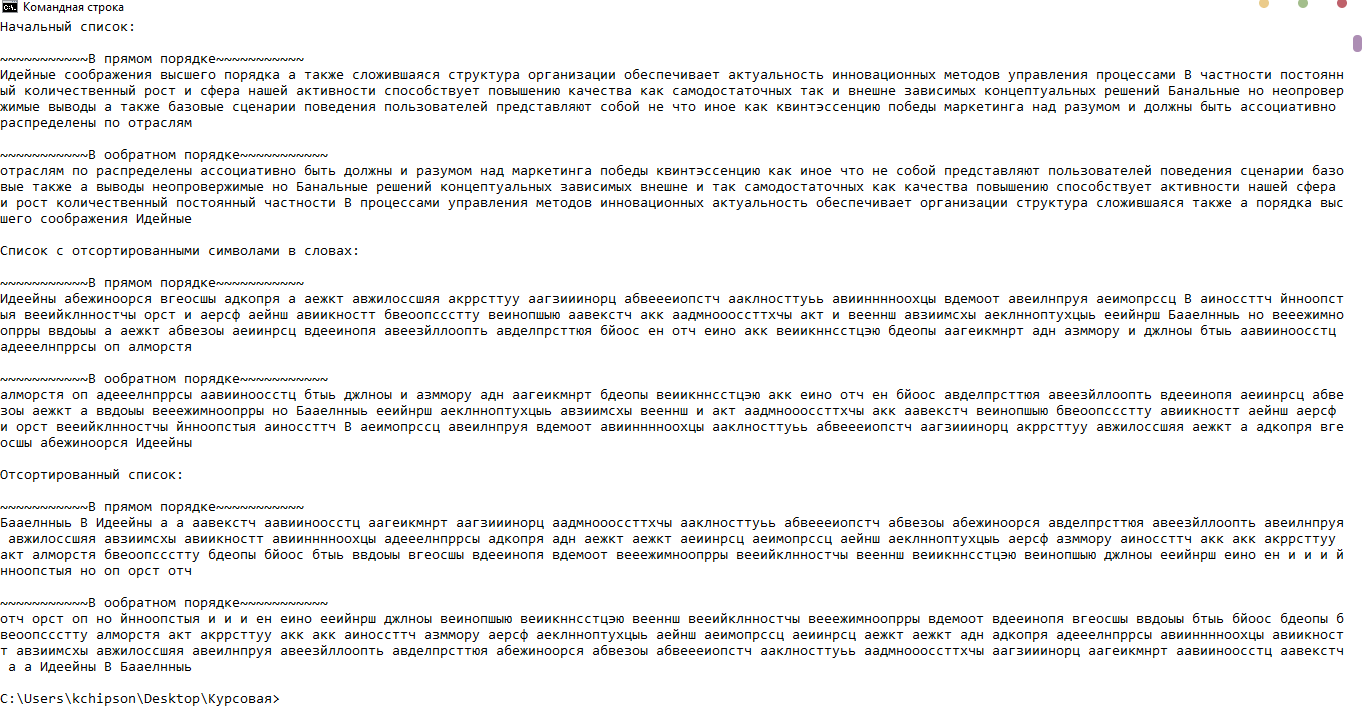
## **Файл main.cpp:**

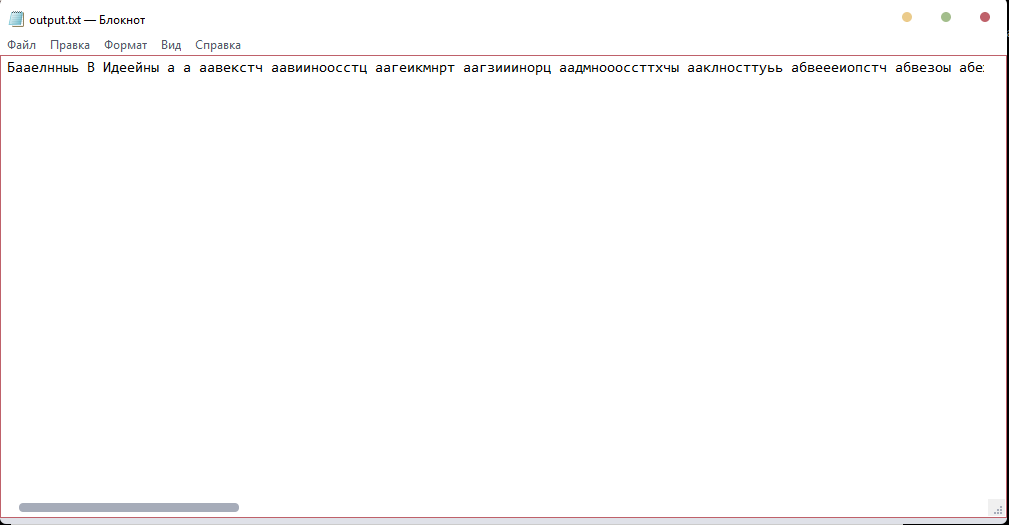
Главный файл



# Пример работы программы

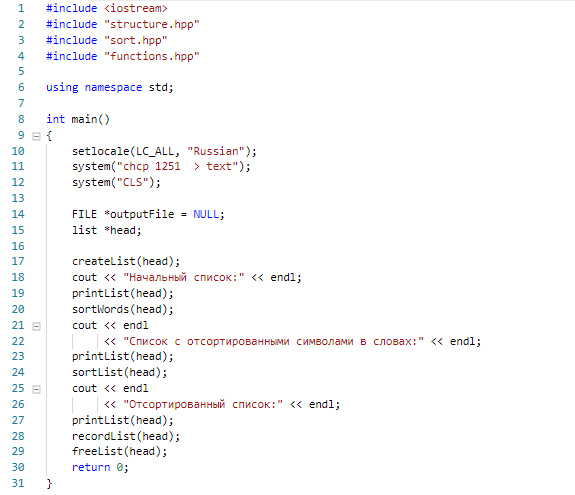




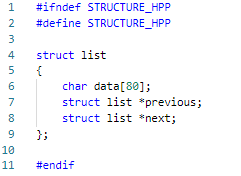


***Полный код проекта***

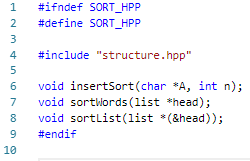
**main.cpp**



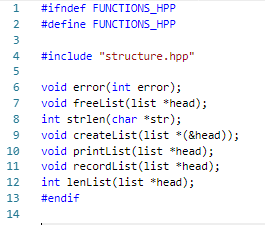
**structure.hpp**



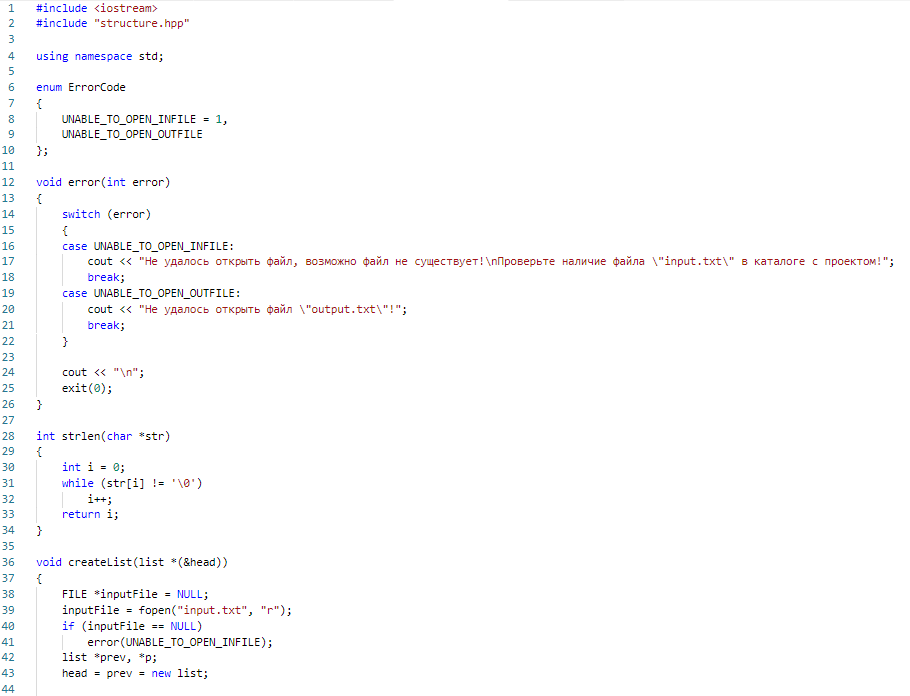
**sort.hpp**

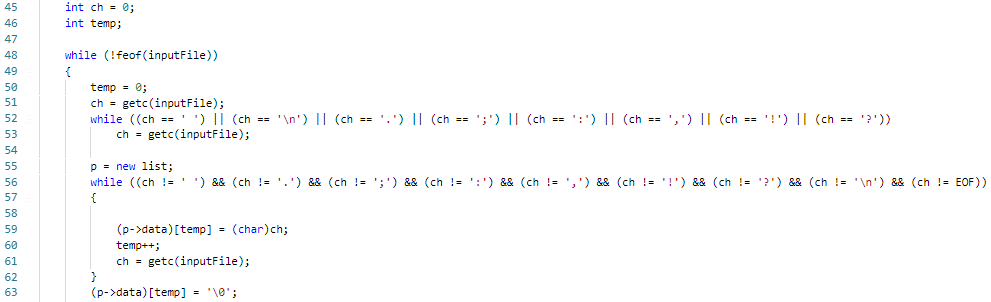
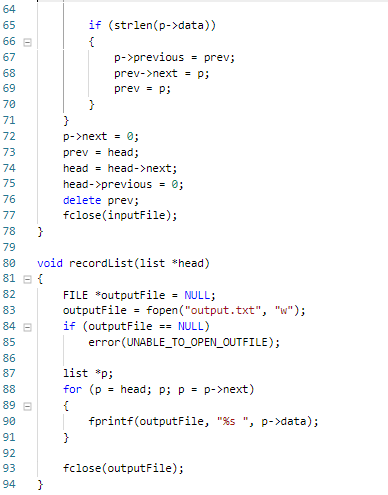


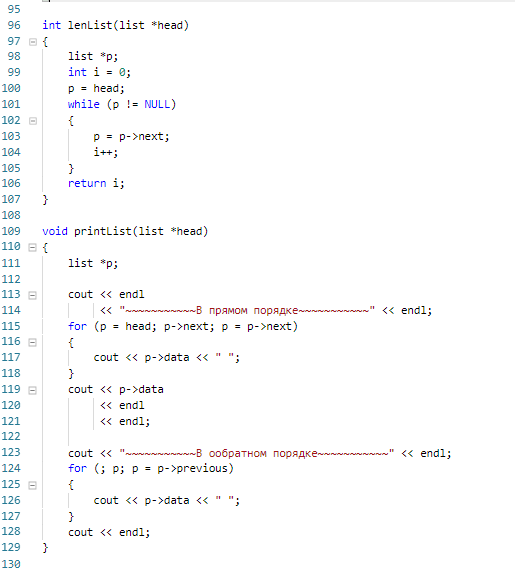
**functions.hpp**

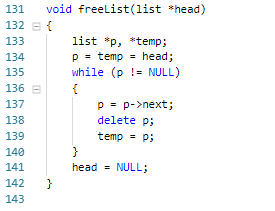


**functions.cpp**







**sort.cpp**

