

ЗАДАНИЕ

Программе на вход подается текст (текст представляет собой предложения, разделенные точкой. Предложения - набор слов, разделенные пробелом или запятой, слова - набор латинских букв, и цифр. Длина текста и каждого предложения заранее не известна.

Программа должна сохранить этот текст в динамический массив строк и оперировать далее только с ним.

Программа должна найти и удалить все повторно встречающиеся предложения (сравнивать их следует посимвольно, но без учета регистра).

Далее, программа должна запрашивать у пользователя одно из следующих доступных действий (программа должна печатать для этого подсказку. Также следует предусмотреть возможность выхода из программы):

1. Вывести все предложения в которых все слова состоят только из цифр.
2. Отсортировать предложения по уменьшению количества заглавных букв в нем.
3. Удалить все предложения, в которых какое-то слово встречается 2 и более раза.
4. Преобразовать все предложения так, чтобы каждое вхождение цифры заменялось на соответствующую подстроку (0 - "Zero", 1 - "One", 2 - "Two", и.т.д.).

Все сортировки должны осуществляться с использованием функции стандартной библиотеки. Использование собственных функций, при наличии аналога среди функций стандартной библиотеки, запрещается.

Все подзадачи, ввод/вывод должны быть реализованы в виде отдельной функции.

АННОТАЦИЯ

При выполнении лабораторной работы на языке программирования С была написана программа, позволяющая работать с текстом и содержащая следующие функции:

- сохранения текста в динамический массив предложений;
- удаления одинаковых предложений (без учета регистра);
- вывода измененного текста на экран;
- сортировка предложений по уменьшению количества заглавных букв в нем;
- удаление всех предложений, в которых какое-то слово встречается 2 и более раз;
- замена каждого вхождения цифры на соответствующую подстроку.

SUMMARY

When performing a laboratory assignment in the C programming language, a program was written that allows you to work with text and contains following functions:

- save text to dynamic array of sentences;
- remove re-occurring sentences (case insensitive);
- writing modified text on the screen;
- sorting sentences to reduce the number of capital letters in it;
- deletion of all sentences in which a word occurs 2 or more times;
- replacing each occurrence of a digit with the corresponding substring.;

СОДЕРЖАНИЕ

Введение.....	4
1. Описание программы	4
1.1 Подключенные библиотеки.....	4
1.2 Функция main.....	4
1.3 Функция input_text.....	7
1.4 Функция delete_duble.....	5
1.5 Функция word_is_only_numbers	5
1.6 Функция sort_cap	5
1.7 Функция ddword.....	6
1.8 Функция digits_in_word.....	6
2. Тестирование.....	7
2.1 Тест функции delete_duble.....	7
2.2 Тест функции word_is_only_numbers.....	8
2.3 Тест функции sort_cap.....	8
2.4 Тест функции ddword	9
2.5 Тест функции digits_in_word	9
ЗАКЛЮЧЕНИЕ.....	9
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	9

Введение

Цель работы:

Написать программу для обработки текста на языке программирования Си. Также закрепить полученные знания о основных функциях стандартной библиотеки Си.

1. Описание программы

1.1 Подключенные библиотеки

В ходе работы были подключены:

- Заголовочный файл, содержащий функции ввода и вывода
- Заголовочный файл, который содержит в себе функции, занимающиеся выделением памяти, контролем процесса выполнения программы, преобразованием типов
- Заголовочный файл стандартной библиотеки языка программирования C, который используется для задач, связанных с локализацией.
- Заголовочный файл стандартной библиотеки языка программирования C, содержащий функции для работы с нуль-терминированными строками и различными функциями работы с памятью.
- Заголовочный файл стандартной библиотеки языка программирования C, в котором объявляется макрос препроцессора языка C `assert()`.
- Заголовочный файл стандартной библиотеки языка программирования C, содержащий объявления функций для классификации и преобразования отдельных символов.

1.2 Функция `main`

Функция выводит подсказку для пользователя и считывается контрольное число – код соответствующей функции. Если такой код существует, то вызывается очередная функция, иначе выводится сообщение, что данные некорректны и функция завершает свою работу.

1.3 Функция `input_text`

Функция считывает массивы строк. Создаётся два динамических массива для строк. Выделяем память для массива. Элемент массива по индексу является массивом из букв, который мы получаем обратившись к функции `input_sentence`. Далее мы добавляем память под ещё один элемент массива. Проверяем выделилась ли память с помощью второго динамического массива.

Функция `input_sentence` считывает строку в массив. Создаётся динамический массив. Выделяется память. Посимвольно вводятся символы до точки. После считывания каждого символа добавляется память под ещё один.

1.4 Функция `delete_duble`

Функция проходит по элементам массива в виде строк и, используя функции `cmp_str` и `delete_sentence`, удаляет повторяющиеся предложения.

Функция `cmp_str` сравнивает строки посимвольно без учёта регистра. С помощью стандартной функции проверяем длины строк. Если они равны сверяем их посимвольно. Если все они равны, возвращаем 0, иначе 1.

Функция `delete_sentence` очищает строку и смещает массив строк.

1.5 Функция `word_is_only_numbers`

Функция выводит предложения, слова в которой состоят только из цифр. Функция проходит по массиву строк, проверяя с помощью функции `cmp_numbers` состоит ли строка из слов, являющихся цифрами, печатает верные.

Функция `cmp_numbers` проверяет состоит ли строка из слов, состоящих из цифр. С помощью стандартной функции проверяем длину строки. Далее проверяем посимвольно. Если встречается символ, не являющийся пробелом, запятой или цифрой функция возвращает 0, иначе 1.

1.6 Функция `sort_cap`

Функция сортирует предложения по уменьшению количества заглавных букв в нем. Создаётся структура, состоящая из строки и количества символов в верхнем регистре. Выделяется память под массив структур. В каждый элемент массива записывается строка и количество символов в верхнем регистре этой строки, вычисленных с помощью функции `cap`. Массив сортируется с помощью стандартной сортировки `qsort`. Массив строк перезаписывается в новом порядке и структура очищается.

Функция `cap` подсчитывает количество элементов строки в верхнем регистре. Каждый символ проверяется и если он в верхнем регистре переменная `counter` увеличивается. Возвращает переменную `counter`.

1.7 Функция `ddword`

Функция удаляет все предложения, в которых какое-то слово встречается 2 и более раза с помощью функций `double_word` и `delete_sentence`. Проходит по массиву строк с помощью функции `double_word` проверяет есть ли в предложении повторяющиеся слова и с помощью функции `delete_sentence` удаляет его.

Функция `double_word` принимает от функции `separ` массив слов, полученных из строки и сравнивает их. Если есть одинаковые функция возвращает 1, иначе 0. Потом она очищает память.

Функция `separ` строку на слова. Создаётся два динамический массива для слов. В начале доходим до первого символа слова. Вычисляем оставшуюся длину строки. Выделяем память под массив, состоящий из строки. С помощью стандартной функции копируем его и дальше работаем с новым массивом. Входим в бесконечный цикл. Расширяем память под массив с элементами в виде слов. Проходим символы не относящиеся к слову в строке и записываем слово в элемент массива слов, доходим до следующего слова и цикл повторяется. Цикл и функция завершаются если символ равен ' '.

Функция `delete_sentence` очищает строку и смещает массив строк.

1.8 Функция `digits_in_word`

Функция заменяет каждого вхождение цифры на соответствующую подстроку. Проходит по массиву строк и с помощью функции `replacement` заменяем цифры на подстроки в предложении.

Функция `replacement` проходится по всем символам строки. Если символ является цифрой, он заменяется на подстроку с помощью функции `ger`.

Функция `ger` заменяет цифру на слово. Проверяем длину строки, длину слова, которым заменим цифру. Выделяем память для новой строки, являющейся суммой длин строки и слова. Далее мы сдвигаем символы вправо от цифры на место расширенной памяти, освобождая место для слова. Заменяем цифру словом (слово записано в отдельно созданный массив и имеет нужный индекс равный цифре). Возвращаем новую строку

2. Тестирование

2.1 Тест функции `delete_duble`

Test Input 1:

The following curious history was related to me by a chance railway acquaintance. He was a gentleman more than seventy years of age, and his thoroughly good and gentle face and earnest and sincere manner imprinted the unmistakable stamp of truth upon every statement which fell from his lips. He said: You know in what reverence the royal white elephant of Siam is held by the people of that country. He was a gentleman more than seventy years of age, and his thoroughly good and gentle face and earnest and sincere manner imprinted the unmistakable stamp of truth upon every statement which fell from his lips. The following curious history was related to me by a chance railway acquaintance. The following curious history was related to me by a chance railway acquaintance.

Test Output 1:

The following curious history was related to me by a chance railway acquaintance. He was a gentleman more than seventy years of age, and his thoroughly good and gentle face and earnest and sincere manner imprinted the unmistakable stamp of truth upon every statement which fell from his lips. He said: You know in what reverence the royal white elephant of Siam is held by the people of that country.

Test Input 2:

the wHite elePhAnt was stoLeN. The white elephant was stolen.

Test Output 2:

the wHite elePhAnt was stoLeN.

Test Input 3:

Then I grew calmer and collected my faculties. Then I grew calmer and collected my facultie.

Test Output 3:

Then I grew calmer and collected my faculties. Then I grew calmer and collected my facultie.

Test Input 4:

Belarussia.Belarussian.

Test Output 4:

Belarussia.Belarussian.

2.2 Тест функции word_is_only_numbers

Test Input 1:

Then I grew calmer and collected my faculties. I soon saw my course-for, indeed, there was but the one, course for an intelligent man to pursue. Lat\$e as it was, I flew to New York and got a policeman to conduct me to the headquarters of the detective force. Fortunately I arrived in time, though the chief of the force, the celebrated Inspector Blunt was just on the point of leaving for his home.8383 4534,45. He was a man of middle size and compact frame, and when he was thinking deeply he had a way of knitting his brows and tapping his forehead reflectively with his finger, which impressed you at once with the conviction that you sto\$od in the presence of a person of no common order. The very sight of him gave me confidence and made me hopeful. I stated my errand. It did not flurry him in the least; it had no more visible effect upon his iron self-possession than if I had told him somebody had stolen my dog.

Test Output 1:

8383 4534,45.

Test Input 2:

Scnhldv dfwgw 3454.4354 3475.

Test Output 2:

4354 3475.

Test Input 3:

Vndfwkjvn34fcsv 3454.64.

Test Output 3:

64.

2.3 Тест функции sort_cap

Test Input 1:

This is no ordinary case. Every step must be warily taken; each step must be made sure before the next is ventured. AND secrecy must be observed secrecy profound and aBsOlutE. Speak to no one AbOut the matter, not even the reporters. I will tAke care of them; I will see that they get only what it may suit my Ends to let them know.

Test Output 1:

AND secrecy must be observed secrecy profound and aBsOlutE.

I will tAke care of them; I will see that they get only what it may suit my Ends to let them know.

Speak to no one AbOut the matter, not even the reporters.

This is no ordinary case.

Every step must be warily taken; each step must be made sure before the next is ventured.

Test Input 2:

dljf;jd.HGKjdhsjHBU,dfhKTF.FGBV hjfss43.JHGBLJGLdsm3,dsa4G.

Test Output 2:

HGKjdhsjHBU,dfhKTF.

FGBV hjfss43.

dljf;jd.

2.4 Тест функции ddword

Test Input 1:

Never mind about that. I may, and I may not. We generally gather a pretty shrewd inkling of who our man is by the manner of his work and the size of the game he goes after. We are not dealing with a pickpocket or a hall thief now, make up your mind to that. This property was not 'lifted' by a novice. But, as I was saying, considering the amount of travel which will have to be done, and the diligence with which the thieves will cover up their traces as they move along, twenty-five thousand may be too small a sum to offer, yet I think it worth while to start with that.

Test Output 1:

Never mind about that.

We are not dealing with a pickpocket or a hall thief now, make up your mind to that.

This property was not 'lifted' by a novice.

2.5 Тест функции

Test Input 1:

34532.345fdkhHV 432.

Test Output 1:

dsdfjTwoThreeFourFiveThreeTwo.

ThreeFourFivefdkhHV FourThreeTwo.

ЗАКЛЮЧЕНИЕ

В ходе работы была написана программа для ввода и обработки текста. Все задачи и подзадачи были реализованы в отдельных функциях. Исходя из тестов программы можно сделать вывод, что она работает корректно.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- Cplusplus: URL: <https://cplusplus.com> (дата обращения 15.12.2018)
- Cppstudio: URL: <https://cppstudio.com> (дата обращения 15.12.2018)