

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА



Институт радиоэлектроники и информационных технологий
Кафедра Информатики и систем управления

Лабораторная работа №1 «Списки. Стеки. Очереди.»

ОТЧЕТ по лабораторной работе № 1

по дисциплине
Технологии программирования

РУКОВОДИТЕЛЬ:

(подпись)

Капранов С.Н.
(фамилия, и.,о.)

СТУДЕНТ:

(подпись)

Куликова Е.А.
(фамилия, и.,о.)

18-ИСТ-4
(шифр группы)

Работа защищена «__» _____

С оценкой _____

Нижний Новгород

2019

Содержание

Введение.....	2
1. Цель работы	3
2. Задачи	3
3. Описание алгоритма	3
4. Код программы.....	4
5. Реализация программы	5
Заключение	7
Используемая литература.....	8

					ЛР1 – НГТУ – 18-ИСТ-4 – 908 – 10						
Изм	Лист	№ Докум.	Подпись	Дата	Лабораторная работа №1						
Разраб.	Куликова Е.А.										
Проверил	Капранов С.Н.										
Н. контр.											
Утв.					Лит.			Лист		Листов	
								1		8	
					Каф. ИСУ 18-ИСТ-4						

Введение

Для выполнения первой лабораторной работы на почту группы было выслано задание, в котором каждый студент должен выполнить задачу исходя из номера своего варианта. Мне достался двадцать третий вариант и задача – «Используя список, удалить из текста программы комментарии и поместить их в отдельный текстовый файл.»

					ЛР1 – НГТУ – 18-ИСТ-4 – 908 – 10	Лист
						2
Изм.	Лист	№ докум.	Подпись	Дата		

1. Цель работы

Создать программу, соответствующую требованиям первой лабораторной работы, которая должна принимать на вход код программы, искать в нём комментарии и записывать все найденные комментарии в новый текстовый файл.

2. Задачи

Поставленные задачи:

1. Разработать алгоритм, по которому будет выполняться программа.
2. Написать код, реализующий задание.
3. Протестировать, чтобы убедиться в правильности решения.

3. Описание алгоритма

На вход программы поступает два параметра: имя входного файла и имя выходного. Далее идёт проверка на количество входящих параметров. Открывается файл на чтение, который проверяется на открытие. Затем следует основная решающая часть. Пока на вход приходят новые строки из файла, программа делает следующее:

1. Ищет комментарии типа «//».
2. Проверяет принадлежит ли пришедшая строка к комментарию типа «/**/».
3. Если принадлежит, то ищет знак закрытия комментария «*/».
4. Ищет открывающий знак комментария типа «/**/».

Каждый найденный комментарий сохраняется в список. Цикл завершается после нахождения всех комментариев во входящем файле. После чего файл на чтение закрывается, и открывается файл на запись. Происходит запись списка в файл. Закрывается файл на запись, освобождается память. Программа завершается.

					ЛР1 – НГТУ – 18-ИСТ-4 – 908 – 10	Лист
Изм.	Лист	№ докум.	Подпись	Дата		3

4. Код программы

```
#include <iostream>
#include <fstream>
#include <list>
#include <string>

int main(int argc, char *argv[]) {
    if (argc != 3)
        return -1; // Неверное количество параметров

    std::list<std::string> comments; // Список комментариев
    std::ifstream in; // Файл на чтение
    std::string line; // Строка для чтения

    int num; // Позиция //
    int num2; // Позиция /*
    int num3; // Позиция */
    bool flag = false; // Флаг для /* ... */
    bool flag2 = false; // Флаг для // ... /* ... */

    in.open(argv[1]);
    if (!in.is_open())
        return -2; // Файл на чтение не открыт

    while (getline(in, line)) {
        num = line.find("//");
        while (num > -1) { // Считываем все //
            num2 = line.find("/*");
            num3 = line.find("*/", num2);
            // Для решения №15
            if (((num2 > num && num3 > num) || num2 == -1) && (!flag)) {
                comments.push_back(line.substr(num));
                flag2 = true;
            }
            num = line.find("//", num + 2);
        }
        if (flag2) {
            flag2 = false;
            continue;
        }
        if (flag) {
            num3 = line.find("*/");
            if (num3 > -1) {
                comments.push_back(line.substr(0, num3));
                flag = false;
            }
            else
                comments.push_back(line);
        }
        num2 = line.find("/*");
        while (num2 > -1) { // Для решения №1
            num3 = line.find("*/", num2 + 2);
            if (num3 > -1) {
                comments.push_back(line.substr(num2, num3 - num2));
            }
            else {
                comments.push_back(line.substr(num2));
                flag = true;
            }
            num2 = line.find("/*", num2 + 1);
        }
    }
}
```

```

        in.close();
        std::ofstream out;
        out.open(argv[2]);
        if (!out.is_open())
            return -3; // Файл для записи не открыт

        for (std::string n : comments) {
            out << n << std::endl;
        }
        out.close();
        comments.clear();

        return 0;
    }

```

Листинг 1 – Код программы

5. Реализация программы

Приходящий на вход текстовый файл «in.cpp», используемый как пример для проверки кода, выглядит следующим образом.

```

#include <stdio.h> /* первый комментарий */ #include <iostream> /* второй комментарий */
/* третий комментарий */
#include <locale.h> /*
четвёртый комментарий
*/ #include <windows.h> /*
пятый комментарий
*/
using namespace std;

int main(int argc, char* argv[])
{
    setlocale(LC_CTYPE, "Russian");
    cout << "8 % 4 = " << 8 % 4 << endl; // шестой комментарий/* седьмой комментарий */
    // восьмой комментарий
    /* девятый комментарий // десятый комментарий /* одиннадцатый комментарий */ int k; /* // двенадцатый комментарий // */
    int m; /* тринадцатый комментарий // */ int a; /*
четырнадцатый комментарий // пятнадцатый комментарий
//
// шестнадцатый комментарий */ int l; /*
//
*/ int n;
// семнадцатый комментарий // восемнадцатый комментарий //
cout << "7 % 4 = " << 7 % 4 << endl; // девятнадцатый комментарий
cout << "2 % 4 = " << 2 % 4 << endl; /* двадцатый комментарий*/
cout << "43 % 10 = " << 43 % 10 << endl;
system("pause");
return 0;
}

```

Рисунок 1 – Код входящего файла

Текстовый файл, куда должны заноситься комментарии, после выполнения программы выглядит следующим образом.

```

/* первый комментарий
/* второй комментарий
/* третий комментарий
/*
четвёртый комментарий
:
/*
пятый комментарий
:
// шестой комментарий/* седьмой комментарий */
// восьмой комментарий
/* девятый комментарий // десятый комментарий /* одиннадцатый комментарий
/* одиннадцатый комментарий
/* // двенадцатый комментарий //
/* тринадцатый комментарий //
/*
:
:    четырнадцатый комментарий // пятнадцатый комментарий
:    //
:    // шестнадцатый комментарий
/*
:    //

// семнадцатый комментарий // восемнадцатый комментарий //
// восемнадцатый комментарий //
//
// девятнадцатый комментарий
/* двадцатый комментарий

```

Рисунок 1 – Код выходящего файла

Повторение некоторых комментариев происходит из-за того, что они являются комментариями комментариев, поэтому записывается изначально комментарий целиком, а после вложенный комментарий.

Заключение

При выполнении работы был разработан алгоритм выполнения поставленной задачи, реализован в виде кода на языке C++ и протестирован.

Используемая литература

1. Справка по C++. List – <https://ru.cppreference.com/w/cpp/container/list>
2. METANIT.COM Сайт о программировании. Контейнер List. – <https://metanit.com/cpp/tutorial/7.6.php>
3. Примеры кода C++ – <http://rubukkit.org/threads/gotovye-programmy-c-obnovljaetsja-reguljarno.55044/>

					ЛР1 – НГТУ – 18-ИСТ-4 – 908 – 10	Лист
						8
Изм.	Лист	№ докум.	Подпись	Дата		