|  |
| --- |
| ГУАП |

КАФЕДРА № 14

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | Н.Ю.Чумакова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №9 |
| Работа с файлами |
| по курсу: Основы программирования |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 1941 |  |  |  | А.А.Коваленко |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2020

**1.Цель Работы:**

Скопировать содержимое текстового файла, ограничив длину строки N символами. Слова, не помещающиеся в строку заданной длины, скопировать в виде новой строки.

**2.Задание:**

Скопировать содержимое текстового файла, ограничив длину строки N символами. Слова, не помещающиеся в строку заданной длины, скопировать в виде новой строки.

**3.Формализация задачи:**

Создаем текстовые файлы "Start.txt" и "End.txt". В "Start.txt" записываем наш тестовый пример. Вводим ограничение в N символов. Считываем строку, которую надо обработать из файла "Start.txt". Если на N символе - разделитель, то переносим все что после него на новую строку. Если же на N символе у нас слово, то находим его начало, создаем новую строчку в файле и записываем.

Вставляем в "End.txt" N символов из строки. Добавляем в этот файл символ конца строки для перехода на новую строку. Записываем оставшиеся символы в новую строку.

В исходном файле не должно быть пустых строчек, отсутствовала Табуляция

**4. Тестовые примеры**

|  |  |
| --- | --- |
| Пример 1 | Пример 2 |
| Строка: Привет, как твои дела  N = 10 | Строка: Как погода как друзья  N = 13 |
| Привет,  как твои дела | Как погода  как друзья |

**5. Схема Алгоритма**

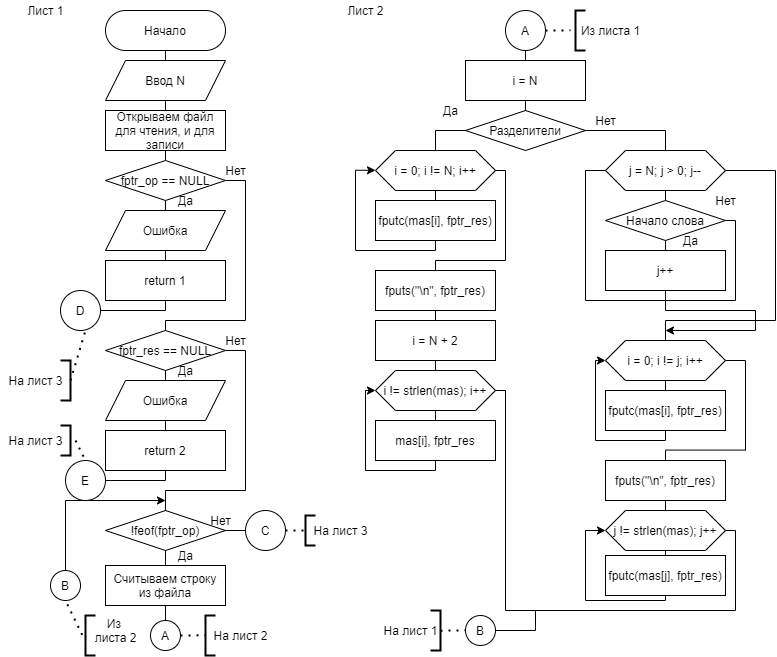
****

Рисунок 1

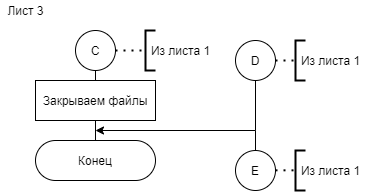


Рисунок 2

**6. Листинг Программы:**

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <locale.h>  #define MAXLINE 500  int main(void)  {  setlocale(LC\_ALL, "Russian");  int N;  char mas[MAXLINE];  printf("Введите N>");  scanf\_s("%d", &N);  FILE\* fptr\_op;  FILE\* fptr\_res;  fopen\_s(&fptr\_op, "C:\\Users\\user\\Desktop\\ДЗ\\Лабы по ОП\\Lab 9\\Start.txt", "r");  fopen\_s(&fptr\_res, "C:\\Users\\user\\Desktop\\ДЗ\\Лабы по ОП\\Lab 9\\End.txt", "w+");  if (fptr\_op == NULL)  {  printf("Error open!\n");  return 1;  }  if (fptr\_res == NULL)  {  printf("Error write!\n");  return 1;  }  printf("Сейчас начнется обработка файла\n");  system("pause");  while (!feof(fptr\_op))  {  fgets(mas, MAXLINE, fptr\_op);  int i = N;    if (mas[i] == ' ' || mas[i] == ',' || mas[i] == ';' || mas[i] == ':')  {  for (i = 0; i != N; i++)  {  fputc(mas[i], fptr\_res);  }  fputs("\n", fptr\_res);  i = N + 2;  for (;i != strlen(mas); i++)  {  fputc(mas[i], fptr\_res);  }  }    else  {  int j;  for (j = N; j > 0; j--)  {  if (mas[j] == ' ')  {  j++;  break;  }  }  for (i = 0; i != j; i++)  {  fputc(mas[i], fptr\_res);  }  fputs("\n", fptr\_res);  for (;j != strlen(mas); j++)  {  fputc(mas[j], fptr\_res);  }  }  }  system("cls");  printf("Файл успешно обработан!\n");  fclose(fptr\_op);  fclose(fptr\_res);  return 0;  } |
|  |

**7.Доказательство правильности работы программы:**

Выполнено в OC Windows 8.1; Среда - Консоль, Блокнот

**Пример 1.**

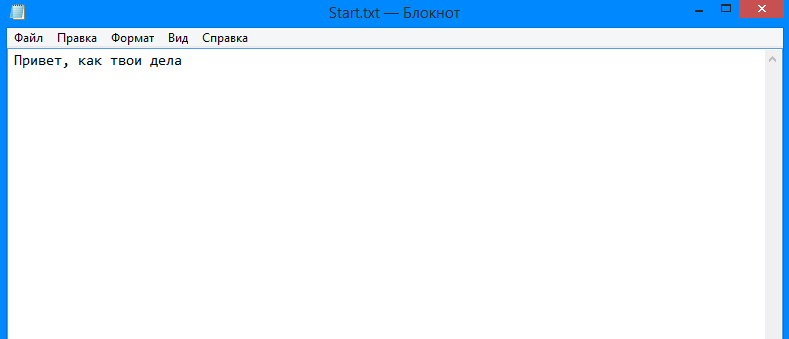


Рисунок 3

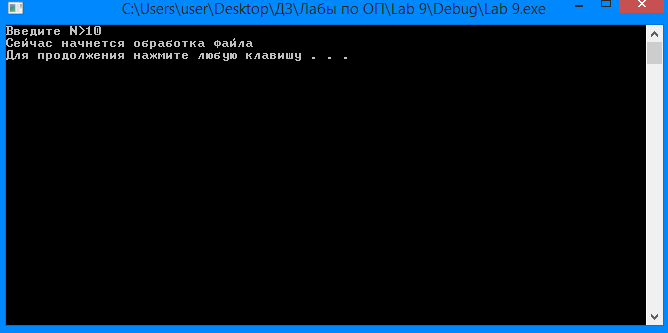


Рисунок 4

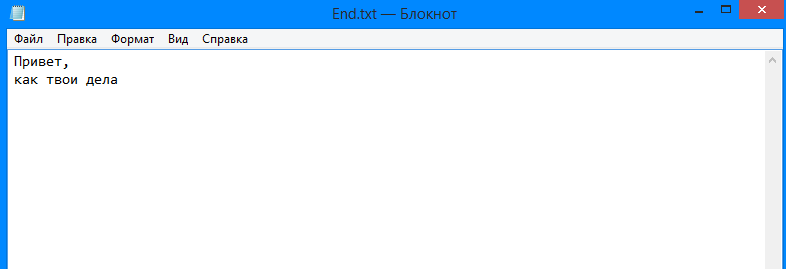


Рисунок 5

**Пример 2**

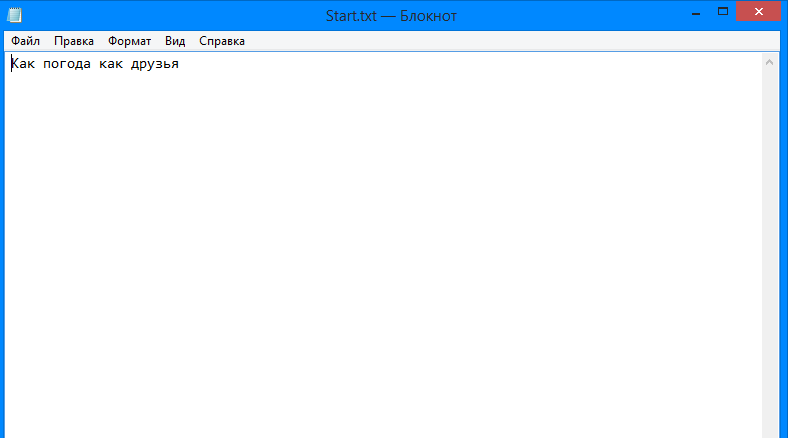


Рисунок 6

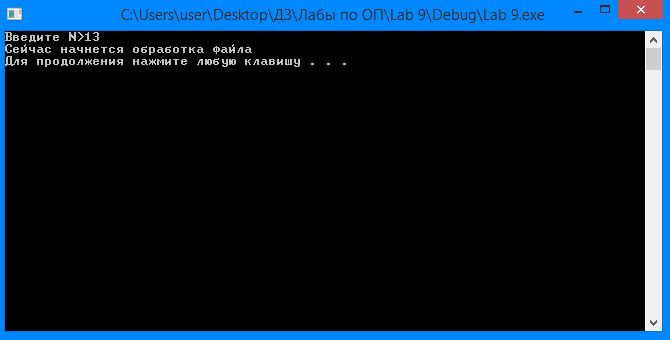


Рисунок 7

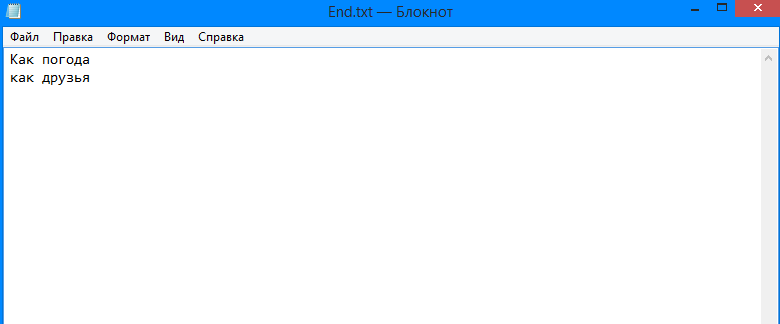


Рисунок 8

**8.Вывод:**

В результате выполнения Лабораторной работы №9 была написана программа, выполняющая задание: Скопировать содержимое текстового файла, ограничив длину строки N символами. Слова, не помещающиеся в строку заданной длины, скопировать в виде новой строки.

Для ее выполнения нам понадобилось создать файл "Start.txt" в который была записана строка для обработки. В консоль мы вводили ограничение N. Результат выводили в "End.txt". Сравнив ручное тестирование с результатом, записанным в блокнот, я делаю вывод, что программа работает исправно