Московский авиационный институт

(Национальный исследовательский университет)

Факультет №3

Кафедра 304

Отчёт по лабораторной работе

По учебной дисциплине

Программирование на языке высокого уровня

На тему:  
“Работа с двоичными файлами”

Выполнил:

Плотников Кирилл

Крещук Николай

Группа: М30-224БК-17

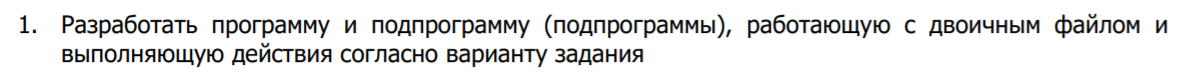
Бригада №8

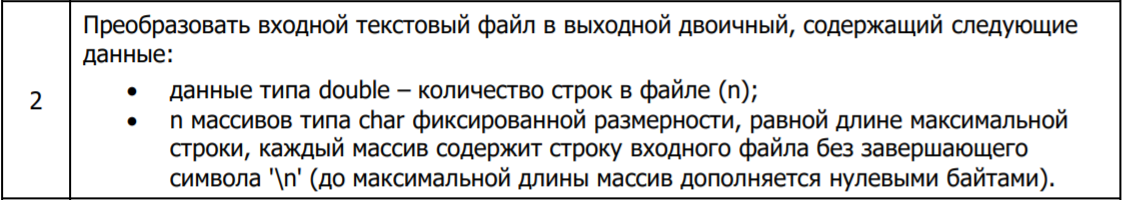
Принял:

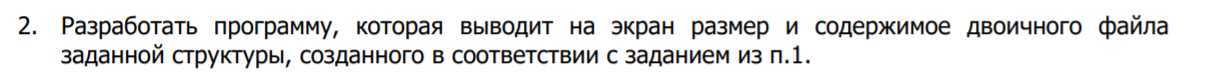
Роговцев А. А

Москва, 2018

**Задание**:

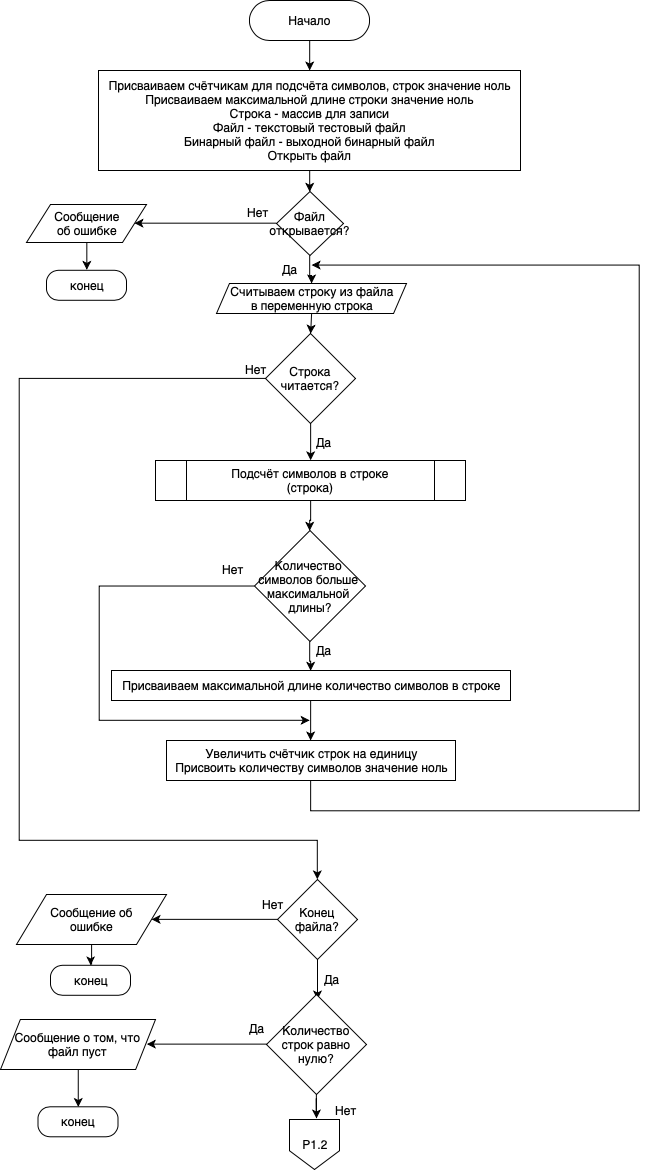


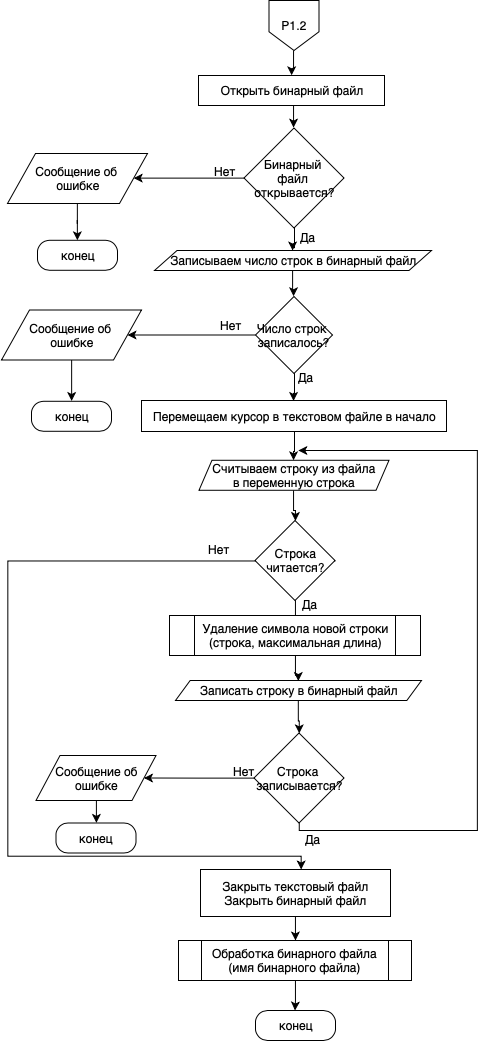


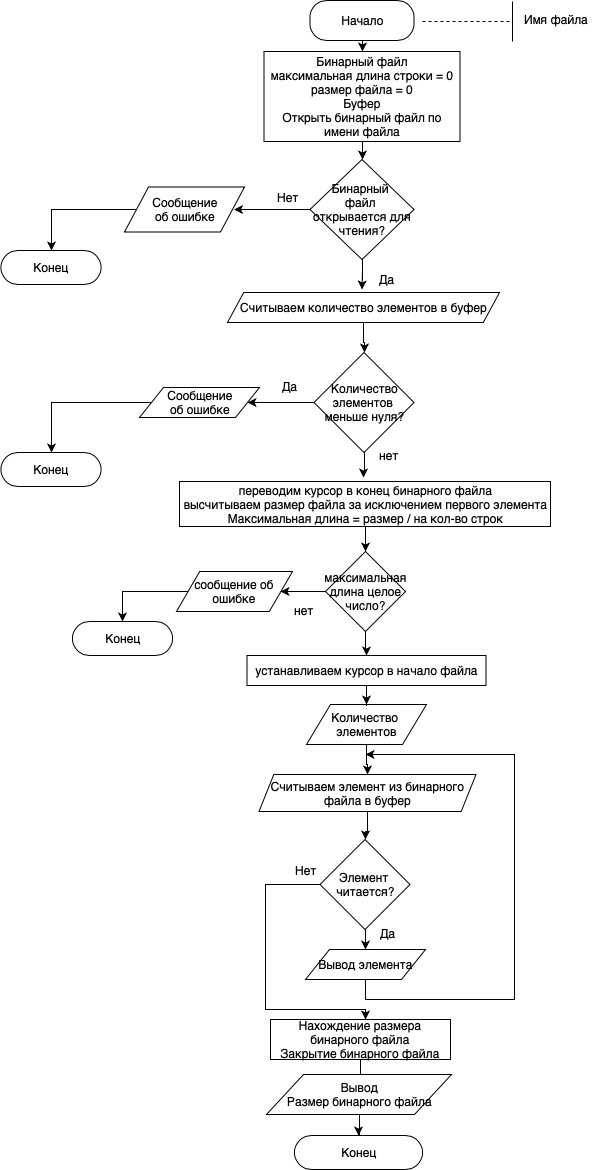


**Структурная схема алгоритма**

**Функция main**

****

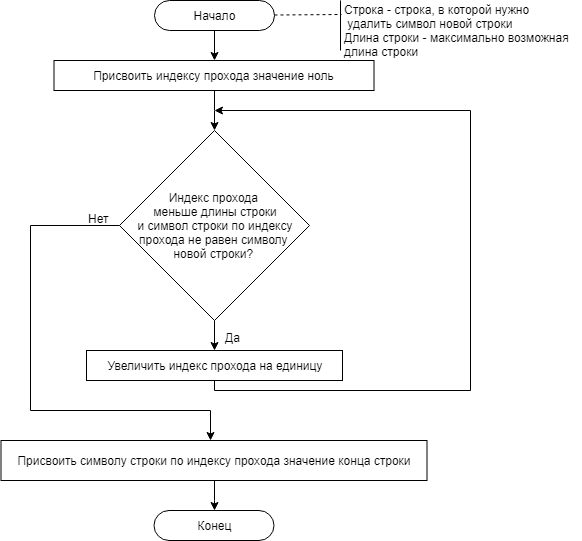
****

**Функция обработки бинарного файл**

**Функция подсчёта длина строки**

****

**Функция удаления символа новой строки**

****

**Текст программы**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Project Name :Lab\_9 \*

\* Project Type :Win32 Console Application \*

\* File Name :Lab\_9.cpp \*

\* Language :C, MAVS 2010 and above \*

\* Programmer(2) :Крещук Николай, Плотников Кирилл, 8 бригада \*

\* Created :20/09/18 \*

\* Last revision :26/10/18 \*

\* Comment :Работа с двоичными файлами \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "pch.h"

#include "stdio.h"

#include <Windows.h>

int CountSymbols(char \*);

void DeleteSymbolNewLine(char \*, int);

void ProcessingBinaryFile(char\*);

int main()

{

// Подключение русского языка.

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

// Максимальная длина строки.

int maxLength = 0;

int countSymb = 0;

double nLine = 0;

FILE \*file;

FILE \*binaryFile;

char line[256];

if ((file = fopen("Test.txt", "r")) == NULL)

{

printf("Ошибка: не удалось открыть файл.\n");

return 2;

}

while (fgets(line, sizeof(line), file) != NULL)

{

countSymb = CountSymbols(line);

if (countSymb > maxLength)

maxLength = countSymb;

nLine++;

countSymb = 0;

}

if (!feof(file))

{

printf("Ошибка: неудачное чтение из файла.\n");

return 3;

}

if (nLine == 0)

{

printf("Ошибка: файл пуст.");

return 4;

}

if ((binaryFile = fopen("BinaryFile.bin", "wb")) == NULL)

{

printf("Ошибка: неудачное открытия файла для записи.\n");

return 3;

}

if (fwrite(&nLine, sizeof(double), 1, binaryFile) != 1)

{

printf("Ошибка: неудачная запись в файл.\n");

return 2;

}

rewind(file);

while (fgets(line, maxLength + 1, file) != NULL)

{

DeleteSymbolNewLine(line, maxLength);

if (fwrite(line, sizeof(char), maxLength, binaryFile) != maxLength)

{

printf("Ошибка: неудачная запись в файл.\n");

return 2;

}

}

fclose(file);

fclose(binaryFile);

ProcessingBinaryFile("BinaryFile.bin");

return 0;

}

// Подсчёт количества символов в строке.

int CountSymbols(char \*line)

{

int nSymb = 0;

// Пока не конец строки.

while (line[nSymb] != '\0')

nSymb++;

return nSymb;

}

void DeleteSymbolNewLine(char \*line, int maxLength)

{

int j = 0;

// Удаление знака новой строки ('\n').

while (j < maxLength && line[j] != '\n')

j++;

line[j] = '\0';

}

void ProcessingBinaryFile(char \*name)

{

FILE\* binaryFile;

int maxLength = 0;

double nLine = 0;

double size = 0;

char line[256];

if ((binaryFile = fopen(name, "rb")) == NULL)

{

printf("Ошибка: неудачное чтение из файла.\n");

return;

}

if (fread(&nLine, sizeof(double), 1, binaryFile) != 1)

{

printf("Ошибка: неудачное чтение из файла.\n");

return;

}

if (nLine < 0)

{

printf("Ошибка: количество элементов не может быть меньше 0.");

return;

}

else

{

fseek(binaryFile, 0, SEEK\_END);

size = ftell(binaryFile) - sizeof(double);

maxLength = size / nLine;

if (size / nLine != (int)(size / nLine))

{

printf("Ошибка: несоответсвие в структуре двоичного файла.\n");

return;

}

fseek(binaryFile, sizeof(double), SEEK\_SET);

}

printf("Количество элементов: %.3f.\n", nLine);

while (fread(line, sizeof(char), maxLength, binaryFile) == maxLength)

printf("%s\n", line);

if (!feof(binaryFile))

{

printf("Ошибка: неудачное чтение из файла.\n");

return;

}

size = ftell(binaryFile);

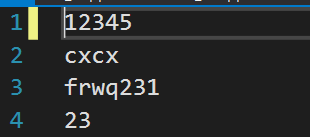
fclose(binaryFile);

printf("Размер файла: %.3f байт.", size);

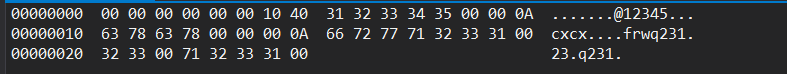
}

**Результат работы программы:**

1) Текстовый файл:



2) Бинарный файл:



3) Прочитанные данные из бинарного файла:

