Национальный исследовательский университет «МИЭТ»

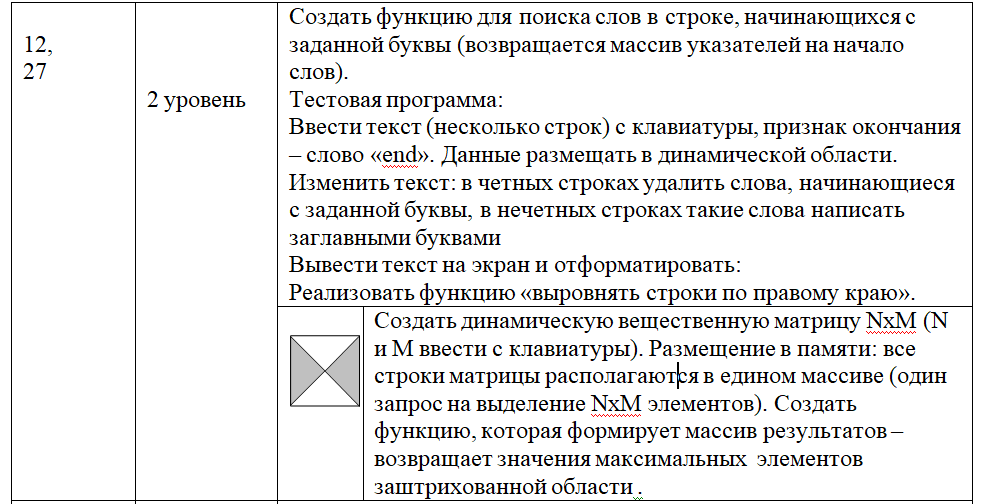
Отчет о выполненной лабораторной работе №5

По предмету :Практикум по программированию на языке С

Выполнила Марина Алина

Группа ПИН-14

2020г.



ПРОГРАММА №1

#include <iostream>

#include <stdio.h>

#include <string.h>

#include <clocale>

#include<iomanip>

char\*\* words(char\*\* List, int N, char ch)// возвращает указатель на начало слова

{

char\* wordsV[256];// рабочий список для ссылок на начала слов

int WCount = 0; // счетчик найденный слов

for (int i = 0; i < N; i++)

{

int fw = 1; // Флаг начала слова

for (char\* uk = List[i]; \*uk != 0; uk++)

if (fw == 1)

{

if (\*uk == ' ') continue;

if (\*uk == ch)

wordsV[WCount++] = uk; // Добавили указатель на слово в вектор указателей

fw = 0; // Сброс флага начала слова

}

else

if (\*uk == ' ')

fw = 1;

}

char\*\* WList = new char\* [WCount + 1];// финальный список ссылок на начала слов

WList[WCount] = NULL; // Признак конца списка

for (int i = 0; i < WCount; i++)

WList[i] = wordsV[i];

return WList;

}

int main()

{

setlocale(0, "Russian");

const int N = 100;// максимальное количество строк

char str[256]; // Вводимая строка

char\*\* strV = new char\* [256]; // Указатель на начало списка

char\*\* strUk = strV; // Указатель, пробегающий по списку

char\*\* strVFin; // Вектор указателей после обработки (NULL - признак окончания списка строк)

int len; // Длина списка слов

printf\_s("Введите букву , которую мы ищем \n");

char ch;

scanf\_s("%c", &ch);

printf\_s("Введите строки \n");

for (len = 0;; len++)

{

gets\_s(str);

if (!strcmp(str, "end")) break; // Сравнение строк

\*strUk = new char[strlen(str) + 1];

strcpy\_s(\*strUk, strlen(str) + 1, str);

strUk++;

}

strVFin = new char\* [len];// создаем чистовой список строк

for (int i = 0; i < len; i++)

strVFin[i] = strV[i];

char\*\* wordsV = words(strVFin, len, ch);// m- та буква которую мы ищем

printf("\n");

char\*\* Wuk = wordsV;

for (Wuk = wordsV; \*Wuk != NULL; Wuk++); // Довести до конца списка слов

Wuk--; // Перейти к последнему элементу списка слов

printf("-----------------------\n");

for (int i = len - 1; Wuk >= wordsV && i >= 0; i--)// проход по списку слов от конца к началу

{

for (; Wuk >= wordsV && \*Wuk >= strVFin[i] && (int)(\*Wuk - strVFin[i]) < (int)strlen(strVFin[i]); Wuk--)

// проход по списку слов от конца к началу

// продолжаем если указатель \*uk указывает на ячейку в памяти внутри строки strVFin[i]

if (i % 2 == 0) // Большие буквы

{

for (char\* uk = \*Wuk; \*uk != ' ' && \*uk != 0; uk++)

if (\*uk >= 'a')\*uk -= 'a' - 'A';

}

else // Удаление слов

{

int k;

for (k = 0; \*(\*Wuk + k) != ' ' && \*(\*Wuk + k) != 0; k++); // Подсчет длины слова

char\* uk;

for (uk = \*Wuk + k; \*uk != 0; uk++)

\*(uk - k) = \*uk;

\*(uk - k) = \*uk; // Копирование символа конца строки

}

}

for (int i = 0; i < len; i++)

{

//std::cout << std::setw(100) << std::right << strV[i] << std::endl;// вывод конечного списка строк по правому краю

//printf("%s\n", strV[i]);// вывод конечного списка строк

printf("%100s\n", strV[i]);// тоже вывод списка

}

}

ПРОГРАММА №2

#include <stdlib.h>

#include <stdio.h>

#include <iostream>

#include<clocale>

#include <ctime>

using namespace std;

double poisk\_maxel1(double\*\* mas, int n, int m)// поиск максимального элемента в левой заштрихованной плоскости

{

double maxel1 = \*\*mas;

for (int j = 0; j < (m / 2); j++)

{

for (int i = j; i < (n - j); i++)

{

if (maxel1 < \*(\*(mas + i) + j))

maxel1 = \*(\*(mas + i) + j);

}

}

return maxel1;

}

double poisk\_maxel2(double\*\* mas, int n, int m)// поиск максимального элемента в правой заштрихованной плоскости

{

double maxel2 = \*(\*(mas + (n / 2 + 1)) + (m / 2 + 1));

for (int j = m / 2; j < m; j++)

{

for (int i = n - j - 1; i <= j; i++)

{

if (maxel2 < \*(\*(mas + i) + j))

maxel2 = \*(\*(mas + i) + j);

}

}

return maxel2;

}

int main()

{

setlocale(LC\_ALL, "Russian");

srand(time(NULL));

int n, m;

cout << " n-это количество строк , m-это количество столбцов" << endl;

cout << "----------------------" << endl;

cout << "Введите n и m " << endl;

cin >> n >> m;

if (n < m)

{

cout << "Ошибка , введите , пожалуйста , еще раз" << endl;

cout << "Количество строк должно быть больше либо равно количеству столбцов" << endl;

//матрица должна походить на прямоугольную или квадратную

//потому что иначе у нее не будет главной и побочной диагонали

// и программма либо совсем не сможет работать, либо будет работать неправильно

}

else

{

double\*\* mas = NULL; // вещественный массив

mas = new double\* [n];

for (int i = 0; i < n; ++i)

{

mas[i] = new double[m];

for (int j = 0; j < m; ++j)

{

mas[i][j] = (double)(rand() % 110);

cout << mas[i][j] << "\t";

}

cout << "\n";

} // создание динамического масссива

cout << "---------------------------" << endl;

double mas1[2];

mas1[0] = poisk\_maxel1(mas, n, m);

mas1[1] = poisk\_maxel2(mas, n, m);

for (int i = 0; i < 2; i++)

cout << mas1[i] << " ";

cout << endl;

delete[]mas;

}

system("pause");

return 0;

}