**Лабораторная работа №5**

|  |  |  |  |
| --- | --- | --- | --- |
| 6 | 2 уровень | Создать функцию для определения одинаковых слов в строке (возвращаются указатели на начало слов).  Тестовая программа:  Ввести текст (несколько строк) с клавиатуры, признак окончания – слово «no». Данные размещать в динамической области. Одинаковые слова в строках написать заглавными буквами. Вывести текст на экран и отформатировать:  Реализовать функцию «выровнять строки по центру». | |
|  | Создать динамическую вещественную матрицу NxM (N и M ввести с клавиатуры). Размещение в памяти: все строки матрицы располагаются в едином массиве (один запрос на выделение NxM элементов). Создать функцию, которая формирует массив результатов – возвращает указатели на нулевые элементы заштрихованной области . |

**Задание 1**

// main.cpp

// lr5\_n1

//

// Created by Дмитрий Джугели on 30.05.2021.

//

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <clocale>

#include <cstdio>

#include <cstdlib>

#include <cstring>

#include <ctime>

#include <cctype>

#include <cmath>

using namespace std;

#define N 100

int odinak\_elements(char\*\* arr, int size\_arr, char\*\*\* MATCH)

{

int t = 1;

char\*\* such = (\*MATCH);

for (int i = 0; i < size\_arr; i++)

{

for (int j = 0; j < size\_arr; j++)

{

if (((strcmp(arr[i], arr[j])) == 0) && (i != j))

{

such = (char\*\*)realloc(such, sizeof(char\*) \* t);

such[t - 1] = arr[i];

t++;

}

}

}

(\*MATCH) = such;

return (t-1);

}

int main()

{

setlocale(0, "Russian");

char ostan[] = "no";

char stroka[N];

char\* arrstrok[N];

char\*\* match = (char\*\*)malloc(3 \* sizeof(char\*));

int a, size\_match;

cout << "Ввод строк\n";

for (a = 0; a < N; a++)

{

gets(stroka);

arrstrok[a] = (char\*)malloc(strlen(stroka) + 1);

strcpy(arrstrok[a], stroka);

if (strcmp(arrstrok[a], ostan) == 0)

break;

}

int v = 0;

size\_match = odinak\_elements(arrstrok, a, &match);

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

{

for (int j = 0; j < size\_match; j++)

{

if (arrstrok[i] == match[j])

v++;

}

if (v >= 1)

{

for (int k = 0; k < strlen(arrstrok[i]) + 1; k++)

arrstrok[i][k] = (char)toupper(arrstrok[i][k]);

}

v = 0;

}

cout << endl << "Вывод отформатированного массива" << endl;

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

{

cout << "\t" << arrstrok[i] << "\t" << endl;

}

// Освобождение памяти

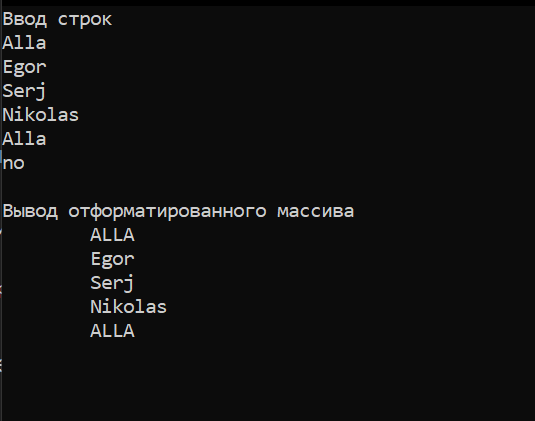
for (int i = 0; i < a; i++) free(arrstrok[i]);

free(arrstrok);

for (int i = 0; i < size\_match; i++) free(match[i]);

free(match);

}



**Задание 2**

|  |  |
| --- | --- |
|  | Создать динамическую вещественную матрицу NxM (N и M ввести с клавиатуры). Размещение в памяти: все строки матрицы располагаются в едином массиве (один запрос на выделение NxM элементов). Создать функцию, которая формирует массив результатов – возвращает указатели на нулевые элементы заштрихованной области . |

//

// main.cpp

// lr5\_n2

//

// Created by Дмитрий Джугели on 1.06.2021.

//

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <clocale>

#include <cstdlib>

#include <ctime>

using namespace std;

int nullobl(double\* osn, int n1, int m1, double\*\*\* nuli)

{

double\*\* jak = (\*nuli);

int l = 0;

for (int i = n1 / 2; i < n1; i++)

{

for (int j = 0; j < m1; j++)

{

if (\*(osn + i \* n1 + j) == 0)

{

jak = (double\*\*)realloc(jak, sizeof(double\*) \* (l + 1));

jak[l] = (osn + i \* n1 + j);

l++;

}

}

}

(\*nuli) = jak;

return l;

}

int main()

{

srand(time(NULL));

setlocale(0, "Russian");

int M, N , t0, t1 , snull;

cout << "Введите N: ";

cin >> N;

cout << "Введите M: ";

cin >> M;

double\* dwumern\_array = (double\*)malloc(sizeof(double) \* N \* M);

double\*\* mass\_null = (double\*\*)malloc(1 \* sizeof(double));

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

{

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

{

t0 = (rand() % 13) - 1;

t1 = (rand() % 10) + 1;

\*(dwumern\_array + i \* N + j) = (double)t0 / (double)t1;

\*(dwumern\_array + 12) = 0;

\*(dwumern\_array + 11) = 0;

cout << \*(dwumern\_array + i \* N + j) << "\t\t\t";

}

cout << endl;

}

snull = nullobl(dwumern\_array, N, M, &mass\_null);

cout << "Количество нулей в этой области: " << snull;

//Высвобождение памяти

for (int i = 0; i < N \* M; i++)

free(dwumern\_array + i);

free(dwumern\_array);

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

free(mass\_null[j]);

free(mass\_null);

}

