Министерство образования и науки России

Федеральное государственное автономное образовательное

учреждение высшего образования

“Санкт-Петербургский государственный электротехнический

университет им. В. И. Ульянова (Ленина)”

(СПБГЭТУ “ЛЭТИ”)

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

**ОТЧЕТ**

**по лабораторной работе № 4 на тему**

**“Указатели языка С / C++” по**

**дисциплине “Программирование”**

**Выполнил:** *Студент группы 5307 Грицай Н. Д.*

**Проверила:** *к.т.н., доцент Сискович Т.И.*

**Подпись преподавателя:\_\_\_\_\_\_\_\_**

Санкт-Петербург

2015

Оглавление

[Цель работы 3](#_Toc433220579)

[1.Задание 3](#_Toc433220580)

[2.Уточнение задания 3](#_Toc433220581)

[3.Контрольные примеры 3](#_Toc433220582)

[4.Описание переменных 4](#_Toc433220583)

[5.Краткое описание алгоритма 4](#_Toc433220584)

[6.Исходный код программы 5](#_Toc433220585)

[7.Результат 8](#_Toc433220586)

[Вывод 8](#_Toc433220587)

# Цель работы

Получить практические навыки использования указателей языка С / C++.

# 1.Задание

Разработать алгоритм и написать программу, выполняющую многократно по желанию пользователя ввод матрицы mxn, контрольный вывод, обработку исходной матрицы и вывод результата.

# 2.Уточнение задания

В программе матрица реализована в виде динамической структуры размерности 2. Результирующая матрица формируется из строк исходной, в которых есть хотя бы 2 совпадающих элемента. В некоторых случаях в результирующей матрице может не оказаться строк.

# 3.Контрольные примеры

Контрольные примеры представлены в таблице 1.

*Таблица 1. Контрольные примеры*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ примера** | **Исходные данные** | | | **Результат** |
| **m** | **n** | **Матрица arr1** | **Матрица arr2** |
| 1 | 1 | 2 | 2 3 | “В матрице нет строк” |
| 2 | 2 | 2 | 1 2  1 1 | 1 1 |
| 3 | 2 | 3 | 2 3 4  1 1 5 | 1 1 5 |
| 4 | 1 | 5 | 1 2 4 535 535 | 1 2 4 535 535 |
| 5 | 2 | 1 | 0  0 | “В матрице нет строк” |
| 6 | 1 | 3 | 0 1 0 | 0 1 0 |

# 4.Описание переменных

Описание переменных представлено в таблице 2.

*Таблица 2. Описание переменных*

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Назначение** |
| arr1 | int\*\* | Указатель на адрес первого элемента исходной динамической структуры |
| arr2 | int\*\* | Указатель на адрес первого элемента результирующей динамической структуры |
| clmct | short | Количество столбцов исходной и результирующей динамических структур |
| str1ct, str2ct | short | Количество строк исходной и результирующей динамических структур |
| isarr1, isarr2 | short | Флажки наличия исходной и результирующей динамических структур(1-есть/0-нет) |
| ismatch | short | Флажок наличия совпадений значений элементов строк исходной динамической структуры(1-есть/0-нет) |
| menu | short | Селектор меню |
| isprogend | short | Флажок выхода из программы(1-выйти/0-нет) |
| i, j, k | short | Вспомогательные переменные |

# 5.Краткое описание алгоритма

1. Вывод меню
2. Выбор пункта меню:

1)Ввод матрицы. Установка флажков isarr1 = 1, isarr2 = 0. Переход в пункт 3

2)Eсли(isarr1), то: вывод исходной матрицы, иначе: сообщение о том, что ее нет. Переход в пункт 3

3)Если(isarr1), то: обработка исходной матрицы и установка флажка isarr2 = 1, иначе: сообщение о том, что ее нет. Переход в пункт 3

4)Если(isarr2), то: вывод результирующей матрицы, иначе: сообщение о том, что ее нет.

Переход в пункт 3

5)Вывод справки. Переход в пункт 3

6)Ввод isprogend. Переход в пункт 3

Если пользователь ввел неправильный пункт меню, то: сообщение об ошибке

1. Если(!isprogend), то: переход в пункт 1
2. Выход из программы

# 6.Исходный код программы

#include "stdafx.h"

#include "Windows.h"

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int \*\*arr1 = NULL, \*\*arr2 = NULL;

short clmct, str1ct, str2ct, ismatch, menu, isarr1 = 0, isarr2 = 0, isprogend = 0;

do

{

system("title 1.4 Динамическая Матрица");

system("cls");

printf\_s("==Меню==\n1.Ввод матрицы.\n2.Вывод исходной матрицы\n3.Обработка

исходной матрицы\n4.Вывод результирующей матрицы\n5.Справка\n

6.Выход\n\nВведите пункт меню: ");

switch (menu = getchar())

{

case '1': //ввод матрицы

if (arr1 != NULL)

{

for (short i = 0; i < str1ct; i++)

free(\*(arr1 + i));

free(arr1);

arr1 = NULL;

}

isarr1 = 1;

isarr2 = 0;

system("cls");

printf\_s("==Ввод матрицы==\n");

//ввод кол-ва строк

do

{

printf("Введите количество строк исходной матрицы: ");

scanf\_s("%hi", &str1ct);

if (str1ct < 1)

printf\_s("Ошибка! Количество столбцов не может быть

меньше 1\n");

} while (str1ct < 1);

//ввод кол-ва столбцов

do

{

printf("Введите количество столбцов исходной матрицы: ");

scanf\_s("%hi", &clmct);

if (clmct < 1)

printf\_s("Ошибка! Количество строк не может быть меньше

1\n");

} while (clmct < 1);

//формирование исходной матрицы

arr1 = (int\*\*)malloc(sizeof(int\*) \* str1ct);

for (short i = 0; i < str1ct; i++)

{

\*(arr1 + i) = (int\*)malloc(sizeof(int) \* clmct);

for (short j = 0; j < clmct; j++)

{

printf\_s("Введите элемент[%hi][%hi]: ", i + 1, j + 1);

scanf\_s("%d", \*(arr1 + i) + j );

}

}

printf\_s("\nВвод завершен!\nДля продолжения нажмите клавишу Enter");

rewind(stdin);

getchar();

break;

case '2': //вывод исходной матрицы

system("cls");

if (isarr1)

{

printf\_s("==Исходная матрица==\n");

for (short i = 0; i < str1ct; i++)

{

for (short j = 0; j < clmct; j++)

printf\_s("%5d", \*(\*(arr1 + i) + j));

printf\_s("\n");

}

}

else

printf\_s("Ошибка! Сначала введите матрицу");

printf\_s("\nДля продолжения нажмите клавишу Enter");

rewind(stdin);

getchar();

break;

case '3': //обработка исходной матрицы

if (arr2 != NULL)

{

for (short i = 0; i < str2ct; i++)

free(\*(arr2 + i));

free(arr2);

arr2 = NULL;

}

system("cls");

if (isarr1)

{

isarr2 = 1;

str2ct = 0;

for (short i = 0; i < str1ct; i++)

{

ismatch = 0;

for (short j = 0; j < clmct - 1 && !ismatch; j++)

for (short k = j + 1; k < clmct && !ismatch;

k++)

if (\*(\*(arr1 + i) + j) == \*(\*(arr1 + i) +

k))

ismatch = 1;

if (ismatch)

{

arr2 = (int\*\*)realloc(arr2, sizeof(int\*) \*

(str2ct + 1));

\*(arr2 + str2ct) = (int\*)malloc(sizeof(int) \*

clmct);

for (short j = 0; j < clmct; j++)

\*(\*(arr2 + str2ct) + j) = \*(\*(arr1 + i) +

j);

str2ct++;

}

}

printf\_s("Результирующая матрица сформирована");

}

else

printf\_s("Ошибка! Сначала введите матрицу");

printf\_s("\nДля продолжения нажмите клавишу Enter");

rewind(stdin);

getchar();

break;

case '4': //вывод результирующей матрицы

system("cls");

if (isarr2)

{

if (str2ct)

{

printf\_s("==Результирующая матрица==\n");

for (short i = 0; i < str2ct; i++)

{

for (short j = 0; j < clmct; j++)

printf\_s("%5d", \*(\*(arr2 + i) + j));

printf\_s("\n");

}

}

else

printf\_s("В результирующей матрице нет ни одной

строки.\n");

}

else

printf\_s("Ошибка! Сначала обработайте исходную матрицу");

printf\_s("\nДля продолжения нажмите клавишу Enter");

rewind(stdin);

getchar();

break;

case '5': //справка

system("cls");

printf\_s("==Справка==\nРезультирующая матрица формируется из строк

новой, в которых есть совпадающие элементы.

\n\nДля продолжения нажмите клавишу Enter");

rewind(stdin);

getchar();

break;

case '6': //выход

system("cls");

printf\_s("Вы уверены, что хотите выйти?(1-Да/0-Нет): ");

scanf\_s("%hi", &isprogend);

rewind(stdin);

break;

default:

printf\_s("Ошибка! Введите существующий пункт меню\n\n");

rewind(stdin);

getchar();

}

} while (!isprogend);

//

if (arr1 != NULL)

{

for (short i = 0; i < str1ct; i++)

free(\*(arr1 + i));

free(arr1);

arr1 = NULL;

}

if (arr2 != NULL)

{

for (short i = 0; i < str2ct; i++)

free(\*(arr2 + i));

free(arr2);

arr2 = NULL;

}

return 0;

}

# 7.Результат

При выполнении программы получены результаты, совпадающие с контрольными примерами, представленными в таблице 1. Ошибки не обнаружены.

# Вывод

В ходе выполнения работы были получены практические навыки использования указателей языка С / C++.