Министерство образования и науки Кыргызской Республики

Кыргызский государственный технический университет

им. И.Раззакова

Факультет информационных технологий

Кафедра «Программное обеспечение компьютерных систем»

отчет

дисциплина «Введение в ПИ»

лабораторная работа №1

Выполнил: студент группы

ИСОП-ПИ(б)-1-17 Ниязбек уулу Эркинбек

Проверила: Профессор Валиева А

Бишкек-2018



Задание №1:

Найти кол-во элементов массива (А=10), находящегося после наименьшего элемента

**Данные:**

Константа n максимального элемента массива.

Array[n] - Массив состоящие из элементов n

min – поиск наименьшего элемента массива

index – индекс наименьшего элемента массива

count – для нахождение после минимального элемента массива

*Тестовый вариант: Дано заданный массив*

* 5 4 3 -7 7 8 9 10 11 12
* Ответ минимум -7 **находящихся после наименьшего элемента 6**

#include <iostream>

# include <conio.h>// библиотека, нужна для использования функций \_kbhit() и \_getch()

using namespace std;

int main()

{

const int n = 10;

int Array[n];

setlocale(LC\_ALL, "Russian");

// ввод массива

cout << "Заполните массив числами.\n";

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

{

cout << "[" << i + 1 << "]: ";

while (!(cin >> Array[i]))

{

cin.clear();

while (cin.get() != '\n');

cout << "Ошибка ввода. Пожалуйста, повторите попытку.\n";

cout << "Введите число.\n";

}

cin.clear();

while (cin.get() != '\n');

system("cls");

}

int min = Array[0];

int index = 0;

// поиск наименьшего элемента

for (int i(1); i < n; i++)

{

if (min >= Array[i])

{

min = Array[i];

index = i;

}

}

// вывод

cout << "Заданный массив: ";

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

{

cout << Array[i] << ", ";

}

cout << "\nНаименьший элемент массива: " << min;

cout << "\nИндекс наименьшего элемента: " << index << endl;

++index;

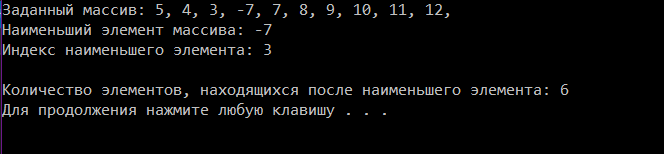
int count = n - index;

cout << "\nКоличество элементов, находящихся после наименьшего элемента: " << count << endl;

system("pause");

return 0;

}





Задание №2: В заданном 2-мерном массиве (А = 3х3) поменять местами

строки, содержащие минимальный и максимальный элемент массива

**Данные:**

n – константа размер массива

Array[n][n] – двух мерный массив состоящие из элементов n

min – поиск минимального элемента массива

max – поиск максимального элемента массива

indexMin – хранение индекса минимального элемента массива

indexMax – хранение индекса максимального элемента массива

a – для временного хранение

* ***Тестовый вариант заданный массив***

1. 2 3

3 2 1

1. 3 4

* Наименьший элемент 1 наибольший элемент 4

*(измененный массив)*

4 3 1

3 2 1

1 2 3

#include <iostream>

#include "conio.h"

using namespace std;

int main()

{

const int n = 3;

int Array[n][n];

setlocale(LC\_ALL, "Russian");

// ввод массива

cout << "Заполните массив целыми числами.\n";

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

for (int j(0); j < n; j++) {

cout << "[" << i + 1 << "][" << j + 1 << "]: ";

while (!(cin >> Array[i][j])) {

cin.clear();

while (cin.get() != '\n');

cout << "Ошибка ввода. Пожалуйста, повторите попытку.\n";

cout << "Введите целое число.\n";

}

cin.clear();

while (cin.get() != '\n');

system("cls");

}

}

int min = Array[0][0];

int max = Array[0][0];

int indexMin = 0;

int indexMax = 0;

//int j = 1;

// поиск минимального и максимального элементов массива и их немер строки

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

for (int j = 0; j < n; j++) {

if (min > Array[i][j]) {

min = Array[i][j];

indexMin = i;

}

if (max < Array[i][j]) {

max = Array[i][j];

indexMax = i;

}

}

}

// проверка на присутствие нескольких равных минимальных/максимальных элементов в массиве

// вывод массива

cout << "Заданный массив:\n";

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

for (int j(0); j < n; j++) {

cout << Array[i][j] << " ";

}

cout << endl;

}

// меняю местами строки содержащие минимальный и максимальный элемент массива

int a;

for (int j(0); j < n; j++) {

a = Array[indexMax][j];

Array[indexMax][j] = Array[indexMin][j];

Array[indexMin][j] = a;

}

// вывод результата

cout << "\nНаименьший элемент массива: " << min << endl;

cout << "Наибольший элемент массива: " << max << endl;

cout << "\nИзмененный массив:\n";

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

for (int j(0); j < n; j++)

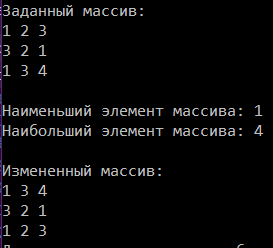
cout << Array[i][j] << " ";

cout << endl;

}

system("pause");

}





Задание №3: Отсортировать массив методом пузырька

n – константа размер массива

Array – массив состоящие из элементов [n]

a - для временного хранение

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL,"ru");

const int n = 10;

system("cls");

int Array[n];

cout << "Заполните массив целыми числами.\n";

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

cout << "Введите значение элемента [" << i + 1 << "]: ";

while (!(cin >> Array[i])) {

cin.clear();

while (cin.get() != '\n') {

system("cls");

cin.clear();

cout << "Поменяйте данные." << endl;

cout << "Введите значение элемента [" << i + 1 << "]:";

}

}

cin.clear();

while (cin.get() != '\n');

system("cls");

}

cout << "Введенный массив: ";

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

cout << Array[i] << " ";

}

cout << endl;

int a;

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

{

for (int i(0); i < n - 1; i++)

{

if (Array[i] >= Array[i + 1])

{

a = Array[i];

Array[i] = Array[i + 1];

Array[i + 1] = a;

}

}

}

cout << "Массив, отсортированный по возрастанию: ";

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

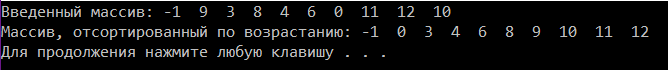
cout << Array[i] << " ";

}

cout<<endl;

system("pause");

return 0;

} 



Задание №4: Отсортировать динамический массив методом пузырька

Функция input для ввода данных

Функция output для вывода данных

Функция sortArray для сортировка массива

\*Array – Динамический массив во всех функцие

\*A – динамический массив на функцие main()

Переменный - n для ввода данных

Переменный – a для временного хранение первого элемента массива

#include <iostream>

using namespace std;

void input(int \*Array, int n) //Функция input для ввода

{

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

{

cout << "Введите значение элемента [" << i + 1 << "]: ";

while (!(cin >> \*(Array + i)))

{

cin.clear();

while (cin.get() != '\n');

cout << "Поменяйте данные." << endl;

cout << "Введите значение элемента [" << i + 1 << "]: ";

}

cin.clear();

//while (cin.get() != '\n');

}

}

void output(int \*Array, int n)

{

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

{

cout << \*(Array + i) << " ";

}

}

void sortArray(int \*Array, int n)

{

int a;

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

{

for (int i(0); i < n - 1; i++)

{

if (\*(Array + i) > \*(Array + i + 1))

{

a = \*(Array + i);

\*(Array + i) = \*(Array + i + 1);

\*(Array + i + 1) = a;

}

}

}

}

int main()

{

setlocale(LC\_ALL, "Russian");

int n;

system("cls");

cout << "Введите размер массива." << endl;

while (!(cin >> n) || n <= 1)

{

cin.clear();

while (cin.get() != '\n');

cout << "Поменяйте данные." << endl;

cout << "Введите размер массива." << endl;

}

int \*A;

A = new int[n];

system("cls");

cout << "Заполение массива.\n";

input(A, n);

system("cls");

cout << "Введенный массив: "; output(A, n); cout << endl;

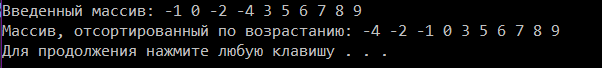
sortArray(A, n);

cout << "Массив, отсортированный по возрастанию: "; output(A, n); cout << endl;

delete[] A;

system("pause");

return 0;}





***Задание №5:2ух мерный динамический массив найти сумму по столбцам***

Функция al – выделение динамический память

Функция output – вывод на экран двух мерного массива

Функция sum – для нахождение суммы столбцов в массиве

Функция rel - освобождение памяти

Переменные – n, m для ввода размера массива

\*\*A – двухмерный массив

matrix = new int \*[n] выделение память состоящие из элементов n

matrix = new int \*[m] выделение память состоящие из элементов m

#include <iostream>

#include <ctime>

#include <stdlib.h>

using namespace std;

using namespace std;

void al(int \*\*&matrix, int n, int m) {

matrix = new int \*[n];

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

matrix[i] = new int[m];

}

}

void output(int \*\*&matrix, int &n, int &m) {

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

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

matrix[i][j] = (0 + rand() % 10);

cout << matrix[i][j] << " ";

}

cout << endl;

}

}

void sum(int \*\*&matrix, int &n, int &m) {

int i, j;

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

int sum = 0;

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

sum += matrix[i][j];

}

cout << "\n [" << i+1 << "]: " << sum << " ";

}

cout << endl;

}

void rel(int \*\*&matrix, int n) {

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

delete[] matrix[i];

}

delete[] matrix;

}

int main() {

setlocale(LC\_ALL, "ru");

srand(time(0));

int n, m;

cout << "Введите n - строк и m - столбцов! ";

cin >> n >> m;

while ((n <= 1) && (m <= 1)) {

cin.clear();

while (cin.get() != '\n');

cout << "Поменяйте данные." << endl;

cout << "Введите размер массива: ";

cin >> n >> m;

cin.clear();

}

int \*\*A = new int \*[n];

al(A, n, m);

output(A, n, m);

sum(A, n, m);

rel(A, n);

system("pause");

return 0;

}



Задача №6: Найти все соверщенные числа в области целых чисел

Данные:

Функция – sum проверка деление без остаток

Переменный – n для ввода данных

Переменный – sum счетчик для соверщенных числы

#include <locale>

#include <iostream>

using namespace std;

int sum(int a)

{

int b = 0;

for (int i = 1; i< a; i++)

{

if (a%i == 0)

b += i;

}

return b;

}

void main()

{

setlocale(LC\_ALL, "Russian");

int n;

m1: cout << "Введите число: "; cin >> n;

if (n >= 0) {

if (n == sum(n))

{

cout << "Результат работы функции для нахождения суммы делителей числа: " << sum(n) << endl;

cout << "Число " << n << " совершенное";

}

else

{

cout << "Число " << n << " не совершенное" << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl;

cout << "Вы ввели не совершенные число \nCовершенные число в интервале " << n << endl;

int sum = 0;

for (int i = 1; i < LONG\_MAX; i++)

{

sum = 0;

for (int j = 1; j < i / 2 + 1; j++)

{

if (i%j == 0)

sum += j;

}

if (sum == i)

cout << "Cовершенное число № " << i << endl;

}

}

}

else {

system("cls");

cout << "Вы ввели не совершенные число!!! введите занова " << endl; goto m1;

}

system("pause");

}