Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 12

По дисциплине «Основы алгоритмизации и программирования»

На тему «Использование указателей»

Выполнил:

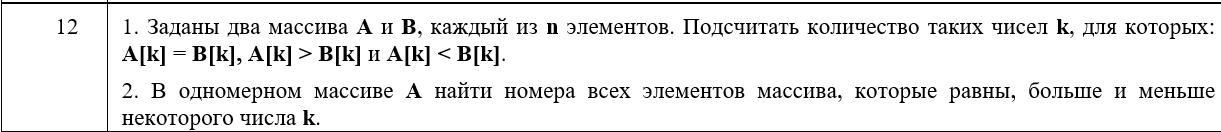
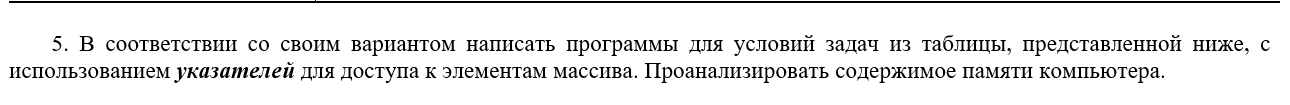
Студент 1 курса 6 группы

Романов Игорь Вячеславович

Преподаватель: асс. Андронова М.В.

2023, Минск

Вариант 12 (основной)



1.

#include <iostream>

#include <ctime> //для работы с функцией time

using namespace std;

//фукнция для заполнения массива A случайными чиислами

void inputArrayA(int\* pA, int size) {

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

\*(pA + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

}

//фукнция для вывода массива A

void outputArrayA(int\* pA, int size) {

cout << "Элементы массива A: ";

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

cout << \*(pA + i) << ", ";

}

cout << endl;

}

//фукнция для заполнения массива B случайными чиислами

void inputArrayB(int\* pB, int size) {

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

\*(pB + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

}

//фукнция для вывода массива B

void outputArrayB(int\* pB, int size) {

cout << "Элементы массива B: ";

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

cout << \*(pB + i) << ", ";

}

cout << endl;

}

//функция для сравнения элементов массива с числом k

void compareElements(int\* pA, int\* pB, int size) {

int counter\_equal = 0, counter\_greater = 0, counter\_less = 0;

for (int k = 0; k < size; k++) { //проверка элементов массива на равенство/неравенство

if (\*(pA + k) == \*(pB + k)) { //расчет равных элементов

counter\_equal++;

}

if (\*(pA + k) > \*(pB + k)) { //расчет элементов массива A больших, чем элементы массива B

counter\_greater++;

}

if (\*(pA + k) < \*(pB + k)) { //расчет элементов массива A меньших, чем элементы массива B

counter\_less++;

}

}

//вывод соответствующих счетчиков на экран

cout << "Случаев A[k] = B[k]: " << counter\_equal << endl;

cout << "Случаев A[k] > B[k]: " << counter\_greater << endl;

cout << "Случаев A[k] < B[k]: " << counter\_less << endl;

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

int A[100], B[100], size; //ввод массива, его размера

cout << "Введите размер массивов size: "; cin >> size;

inputArrayA(A, size); //заполнение массива

int\* pA = A; //создание указателя на массив

outputArrayA(A, size); //вывод элементов массива

inputArrayB(B, size); //заполнение массива

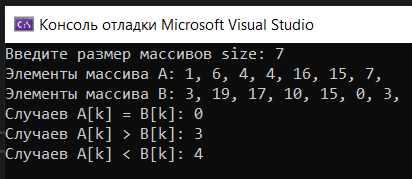
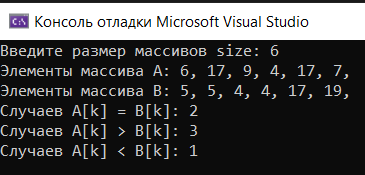
int\* pB = B; //создание указателя на массив

outputArrayB(B, size); //вывод элементов массива

compareElements(A, B, size); //сравнение элементов массивов A и B

return 0;

}

2.

#include <iostream>

#include <ctime> //для работы с функцией time

using namespace std;

//фукнция для заполнения массива случайными числами

void inputArray(int\* pA, int size) {

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

\*(pA + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

}

//фукнция для вывода массива

void outputArray(int\* pA, int size) {

cout << "Элементы массива A: ";

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

cout << \*(pA + i) << ", ";

}

cout << endl;

}

//функция для сравнения элементов массива с числом k

void countElements(int\* pA, int size, int k) {

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

if (\*pA == k) {

cout << "Номер элемента равного k: " << i << endl;

}

else if (\*pA > k) {

cout << "Номер элемента большего, чем k: " << i << endl;

}

else {

cout << "Номер элемента меньшего, чем k: " << i << endl;

}

pA++;

}

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

int A[100], size, k; //ввод массива, его размера и числа k

cout << "Введите число k: "; cin >> k;

cout << "Введите размер массива size: "; cin >> size;

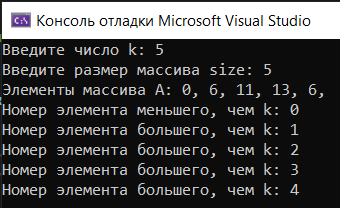
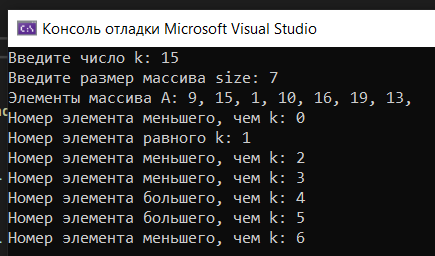
inputArray(A, size); //заполнение массива

outputArray(A, size); //вывод элементов массива

countElements(A, size, k); //расчет элементов массива с числом k

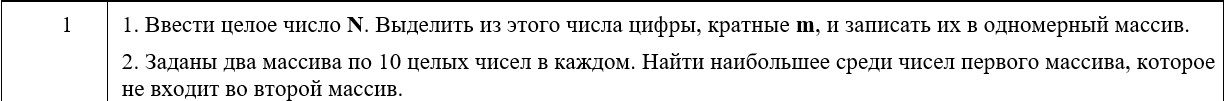
return 0;

}

Дополнительные задания:

Вариант 1



1.

#include <iostream>

using namespace std;

//фукнция для проверки числа N на кратность m

void findMultiples(int N, int m, int\* pArray, int& array\_size) {

array\_size = 0; //ввод размера массива

while (N > 0) {

int digit = N % 10;

if (digit % m == 0) { //проверка числа на кратность

\*pArray = digit; //указатель сохраняет цифру в массиве

pArray++; //увеличиваем ячейку в массиве

array\_size++; //увеличиваем размер массива

}

N /= 10; //удаление последней цифры из числа

}

}

int main() {

setlocale(LC\_CTYPE, "Russian");

int N, m, array[10], array\_size;

cout << "Введите целое число N: "; cin >> N; //ввод числа m

cout << "Введите целое число m: "; cin >> m; //ввод числа n

findMultiples(N, m, array, array\_size); //проверка числа N на кратность m

if (array\_size == 0) { //елси нет кртаных цифр

cout << "Цифры числа N не кратны числу m";

}

else {

cout << "Цифры числа " << N << " кратные " << m << " : "; //если есть кратные цифры

for (int i = array\_size - 1; i >= 0; --i) {

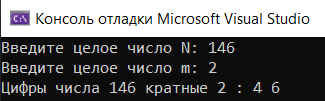
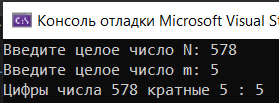
cout << array[i] << " ";

}

}

return 0;

}

2.

#include <iostream>

#include <ctime> //для работы с функцией time

#include <algorithm> //для сортировки массивов

using namespace std;

//функция для заполнения массива A случайными числами и его вывода

void inputArrays(int\* pA, int\* pB) {

for (int i = 0; i < 10; i++) {

\*(pA + i) = rand() % 20; //заполнение массива A случайными числами от 0 до 20

}

sort(pA, pA + 10, greater<>()); //сортировка элементов по убыванию массива A

cout << "Элементы массива A: "; //вывод массива A

for (int i = 0; i < 10; i++) {

cout << \*(pA + i) << ", ";

}

cout << endl;

for (int i = 0; i < 10; i++) {

\*(pB + i) = rand() % 20; //заполнение массива B случайными числами от 0 до 20

}

sort(pB, pB + 10, greater<>()); //сортировка элементов по убыванию массива B

cout << "Элементы массива B: "; //вывод массива B

for (int i = 0; i < 10; i++) {

cout << \*(pB + i) << ", ";

}

cout << endl;

}

//функция, проверяющие вхождение максимального элемента массива A в массив B

void checkMax(int\* pA, int\* pB, int maxA) {

for (int i = 0; i < 10; i++) {

if (\*(pA + i) != \*(pB + i)) {

cout << "Максильмальное число массива A, не входящее в массив B " << \*(pA + i); break;

}

}

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

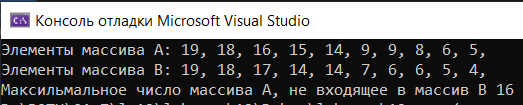
int A[10], B[10], maxA = -1, result; //ввод массивов, их размеров

inputArrays(A, B); //заполнение и вывод массивов

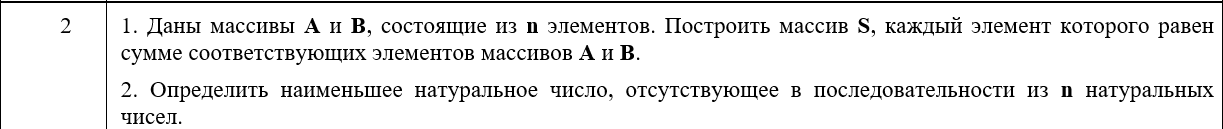
checkMax(A, B, maxA); //проверка элементов массивов

return 0;

}



Вариант 2



1.

#include <iostream>

#include <ctime> //для работы с функцией time

using namespace std;

//фукнция для заполнения массива A случайными числами и его вывода

void arrayA(int\* pA, int n) {

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

\*(pA + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

cout << "Элементы массива A: "; //вывод массива A

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

cout << \*(pA + i) << ", ";

}

cout << endl;

}

//фукнция для заполнения массива B случайными числами и его вывода

void arrayB(int\* pB, int n) {

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

\*(pB + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

cout << "Элементы массива B: "; //вывод массива B

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

cout << \*(pB + i) << ", ";

}

cout << endl;

}

//функции для построения массива S

void buildArray(int\* pA, int\* pB, int\* pS, int n) {

cout << "Массив S: ";

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

\*(pS + i) = \*(pA + i) + \*(pB + i); //присваивание элементам массива S сумме элементов массивов A и B

cout << \*(pS + i) << " "; //вывод массива S

}

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

int A[100], B[100], S[100], n; //ввод массивов, их размеров

cout << "Введите размер массивов size: "; cin >> n;

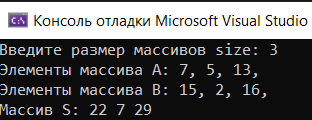
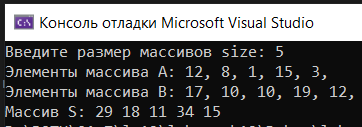
arrayA(A, n); //заполнение массива и его вывод

arrayB(B, n); //заполнение массива

buildArray(A, B, S, n);

return 0;

}



2.

#include <iostream>

#include <ctime> //для работы с функцией time

using namespace std;

//фукнция для заполнения массива A случайными числами и его вывода

void arrayA(int\* pA, int n) {

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

\*(pA + i) = 1+rand() % 20; //заполнение массива случайными числами от 1 до 20

}

cout << "Элементы массива A: "; //вывод массива A

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

cout << \*(pA + i) << ", ";

}

cout << endl;

}

//функция для сравнения элементов массива A с переменной min\_natural

void compareElements(int\* pA, int n, int min\_natural) {

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

if (\*(pA + i) == min\_natural) { //если min\_natural Найдено в массиве, то увеличить его значение на 1

min\_natural++;

}

}

cout << "Наименьшее натуральное число, отсутствующее в последовательности из " << n << " натуральных чисел = " << min\_natural;

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

int A[100], n, min\_natural = 1; //ввод массива, его размера и наименьшего натурального числа

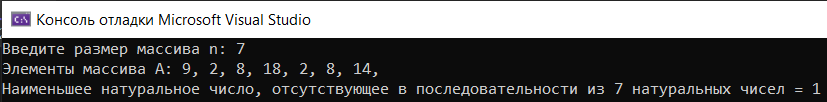
cout << "Введите размер массива n: "; cin >> n;

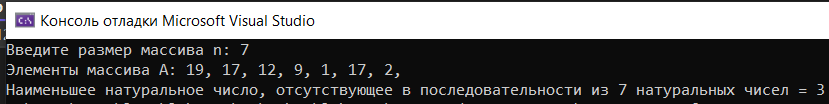
arrayA(A, n); //заполнение массива и его вывод

compareElements(A, n, min\_natural); //сравнение элементов массива с 1

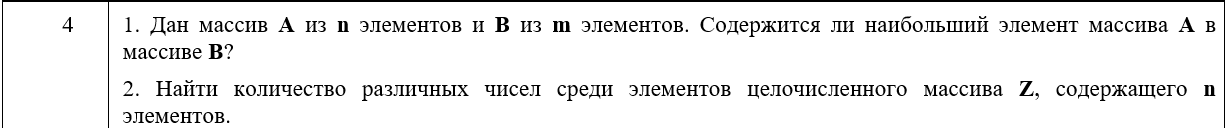
return 0;

}





Вариант 4



1.

#include <iostream>

#include <ctime> //для работы с функцией time

using namespace std;

//фукнция для заполнения массива A случайными числами и его вывода

int arrayA(int\* pA, int n, int maxA) {

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

\*(pA + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

if (\*(pA + i) > maxA) {

maxA = \*(pA + i);

}

}

cout << "Элементы массива A: "; //вывод массива A

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

cout << \*(pA + i) << ", ";

}

cout << endl;

return maxA;

}

//фукнция для заполнения массива B случайными числами и его вывода

void arrayB(int\* pB, int n, int maxA) {

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

\*(pB + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

cout << "Элементы массива B: "; //вывод массива B

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

cout << \*(pB + i) << ", ";

}

cout << endl;

}

//функция на проверку, есть ли в массиве B maxA

int searchMax(int\* pB, int n, int maxA) {

int result = 0;

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

if (\*(pB + i) == maxA) {

result++;

}

}

return result;

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

int A[100], B[100], n, m, maxA = -1, result; //ввод массивов, их размеров

cout << "Введите размер массива A = "; cin >> n;

cout << "Введите размер массива B = "; cin >> m;

maxA = arrayA(A, n, maxA); //вывод максимального элемента массива A

cout << "максимальное число массива A = " << maxA << endl;

arrayB(B, m, maxA); //заполнение массива B

result = searchMax(B, n, maxA); //проверка, есть ли maxA в массиве B

if (result > 0) { //если значение result положительное

cout << "В массиве B есть maxA";

}

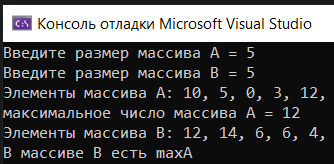
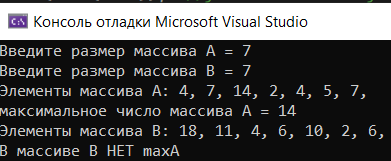
else { //если равно 0

cout << "В массиве B НЕТ maxA";

}

return 0;

}



2.

#include <iostream>

#include <ctime> //для работы с функцией time

using namespace std;

//фукнция для заполнения массива Z случайными числами и его вывода

void arrayZ(int\* pZ, int n) {

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

\*(pZ + i) = rand() % 20; //заполнение массива случайными числами от 0 до 20

}

cout << "Элементы массива Z: "; //вывод массива Z

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

cout << \*(pZ + i) << ", ";

}

cout << endl;

}

int compareElements(int\* pZ, int n, int counter, int flag) {

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

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

flag = 1; // по умолчанию считается, что повторяющихся элементов нет

if (\*(pZ + i) == \*(pZ + j)) {

flag = 0; //если они обнаруживаются, то флажку присваивается 0

}

}

if (flag) { //счетчик увеличивается в случае отсутствия повторяющихся элементов

counter++;

}

}

return counter; //возвращение значения счетчика

}

int main() {

setlocale(LC\_CTYPE, "Russian");

srand(time(NULL)); //устанавливает новую точку отсчета для функции rand()

int Z[100], n, counter = 0, result, flag = 0; //ввод массива, его размера, счетчика повторяющихся элементов, флажка

cout << "Введите размер массива Z = "; cin >> n;

arrayZ(Z, n);

result = compareElements(Z, n, counter, flag); //поиск различных элементов массива

cout << "Количество неповторяющихся элементов: " << result;

return 0;

}

