1)(ДЗ) Дано два масиви: ***А*** [30] і ***B*** [30]. Заповнюємо випадковими числами від 10 до 99. Необхідно створити третій масив ***С***, в якому потрібно зібрати елементи масивів ***A*** і ***B***, які не є загальними для них, без повторень. (обробляти масив через вказівники)

#include <iostream>

#include <ctime>

#include <cstdlib>

#include <algorithm>

using namespace std;

void viewArray(int Array[], int size) {

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

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

}

cout << endl;

}

void chargeArray(int size, int\* ptrA, int\* ptrB, int\* ptrC) {

int count = 0;

int temp\_number;

bool copy;

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

copy = false;

temp\_number = ptrA[i];

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

if (temp\_number == ptrB[i]) {

copy = true;

break;

}

}

if (!copy) {

ptrC[count++] = temp\_number;

}

}

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

copy = false;

temp\_number = ptrB[i];

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

if (temp\_number == ptrA[i]) {

copy = true;

break;

}

}

if (!copy) {

ptrC[count++] = temp\_number;

}

}

cout << "Масив C: ";

viewArray(ptrC, size \* 2);

}

int main() {

const int size = 30;

int A[size];

int B[size];

int C[size \* 2];

srand(time(NULL));

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

A[i] = rand() % 90 + 10;

B[i] = rand() % 90 + 10;

}

int\* ptrA = A;

int\* ptrB = B;

int\* ptrC = C;

cout << "Масив A: ";

viewArray(A, size);

cout << "Масив B: ";

viewArray(B, size);

chargeArray(size, ptrA, ptrB, ptrC);

}



2) Дано ціле число ***K*** і масив ***А***[50] заповнений цілими числами з діапазону від 10 до 99. Вивести індекс першого числа в масиві, більшого за ***K***. Якщо таких чисел немає, то вивести -1. (обробляти масив через вказівники)

#include <iostream>

#include <ctime>

#include <cstdlib>

#include <algorithm>

using namespace std;

void viewArray(int\* ptrA, int size) {

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

cout << ptrA[i] << " ";

}

cout << endl;

}

void kLower(int\* ptrA, int size, int K) {

bool check = false;

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

if (K < ptrA[i]) {

check = true;

cout << "First index number highest K = " << i << endl;

break;

}

}

if (!check) {

cout << "-1" << endl;

}

}

int main() {

const int size = 50;

int A[size];

int K = 98;

srand(time(NULL));

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

A[i] = rand() % 90 + 10;

}

int\* ptrA = A;

cout << "K = " << K << endl;

cout << "Array A: ";

viewArray(ptrA, size);

kLower(ptrA, size, K);

}



Результати роботи програм проілюструвати скріншотами