ОТЧЁТ С ЛАБОРАТОРНОЙ РАБОТЫ #6

за курсом «Програмвання»

студента группы ПА-19-2

Ильяшенко Егора Виталиевича

кафедра компьютерных технологий, ДНУ

2019/2020 уч. год

1. Постановка задачи:

Необходимо было сделать задачу #8 из списка задач. Программа принимает 2 вектора А и В. Треба:

1) визначити вектор С так, щоб усі його елементи були рівні сумі відповідних елементів векторів А і В, якщо в А і В різниця між будь-якими двома сусідніми елементами не перевищує максимального значення двох векторів; якщо ж перевищує, то C буде вектором нульових елементів, кількість яких дорівнюватиме цілій частині максимального значення з двох векторів, якщо вона не перебільшує максимально можливої кількості елементів в C, інакше — кількість дорівнює максимальній кількості нулів у векторах A і B;

2) знайти добуток елементів вектора A, розташованих між першим і другим нульовими елементами;

3) перетворити вектор B так, щоб спочатку розташовувалися всі елементи, що дорівнюють нулю, а потім – усі інші елементи.

1. Описание решение

Послiдовно зробив усi пункти в одному файлi. Вектора записанi у динамiчних масивах i заповнюються тiльки пiсля вводу кол-ва елементiв вектора.

1. Исходный текст программы

#include <iostream>

using namespace std;

void NLO(string text) // Next Line Out

{

cout << endl << text;

}

int main()

{

setlocale(LC\_ALL, "Russian");

cout << "Лабораторная робота №6 (Вариант №8)";

int\* a;

int\* b;

int\* c;

NLO("Введите кол-во элементов первого вектора = ");

int n; cin >> n;

a = new int[n];

NLO("Введите первый вектор (По одному элементу за раз)"); cout << endl;

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

{

cin >> a[i];

cout << endl;

}

NLO("Введите кол-во элементов второго вектора = ");

int m; cin >> m;

if (n <= 0 && m <= 0)

{

NLO("Вы ввели пустые вектора");

return 0;

}

b = new int[m];

NLO("Введите второй вектор (По одному элементу за раз)"); cout << endl;

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

{

cin >> b[i];

cout << endl;

}

int c\_l;

int condition = -1;

if (n > m)

{

c = new int[m];

c\_l = m;

}

else

{

c = new int[n];

c\_l = n;

}

int max\_a\_b;

if (n > 0)

max\_a\_b = a[0];

else if (m > 0)

max\_a\_b = a[m];

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

if (a[i] > max\_a\_b)

max\_a\_b = a[i];

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

if (b[i] > max\_a\_b)

max\_a\_b = b[i];

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

if ((a[i] - a[i - i] > max\_a\_b) || (a[i - 1] - a[i] > max\_a\_b))

condition = 1;

else condition = 2;

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

if ((b[i] - b[i - i] > max\_a\_b) || (b[i - 1] - b[i] > max\_a\_b))

condition = 1;

else condition = 2;

if (condition == 2)

{

for (int i = 0; i < c\_l; i++)

{

c[i] = a[i] + b[i];

}

}

else if (condition == 1)

{

if (max\_a\_b <= n && max\_a\_b <= m)

{

c = new int[max\_a\_b];

for (int i = 0; i < max\_a\_b; i++)

c[i] = 0;

}

else

{

int a\_zero = 0;

int b\_zero = 0;

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

if (a[i] == 0) a\_zero++;

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

if (b[i] == 0)b\_zero++;

if (a\_zero > b\_zero)

c = new int[a\_zero];

else c = new int[b\_zero];

}

}

int dob\_a = 1;

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

{

if (a[i] == 0)

{

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

if (a[j] != 0)

dob\_a \*= a[j];

else break;

break;

}

}

while (1)

{

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

{

if (b[i] - b[i - 1] == -1 \* b[i - 1])

{

int q = b[i];

b[i] = b[i - 1];

b[i - 1] = q;

}

}

int ind = -1;

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

{

if (b[i] == 0 && b[i-1] == 0) ind = i+1;

}

bool end = true;

for (int i = ind; i < m; i++)

{

if (b[i] == 0) end = false;

}

if (end) break;

}

NLO("Добуток элементов между первым и вторым нулевым элементом вектора А = "); cout << dob\_a;

cout << endl << endl;

NLO("Изменённый вектор B = ");

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

cout << b[i] << endl;

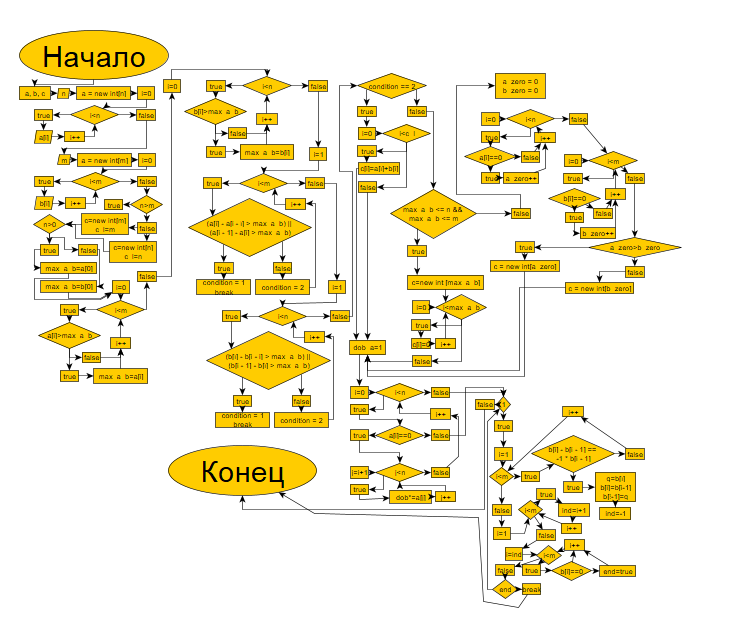
return 0;

}

1. Описание интерфейса

После запуска программы пользователю необходимо ввести кол-во элементов вектора А и последовательно по одному элементу. После чего ввести второй вектор по тому же принципу. После этого программа выведет результат.

1. Блок схема



1. Вывод

