**Ma’lumotlarning abstrak turlari va ma’lumotlar strukturalari. (Vektor, Ro’yxat)**

**3 – Topshiriq**

*Iqbolshoh Iqbolshoh*

2. Vektor hosil qiling size () va empty() funksiyasidan foydalaning va vektor uzunligini aniqlang.

#include <iostream>

#include <vector>

using namespace std;

int main() {

  // Vektor hosil qilish

  vector<int> V;

  // Element qo'shish

  V.push\_back(10);

  V.push\_back(20);

  V.push\_back(30);

  // empty() funksiyasidan foydalanish

  if (V.empty()) {

    cout << "Vektor bo'sh" << endl;

  } else {

    cout << "Vektor bo'sh emas" << endl;

  }

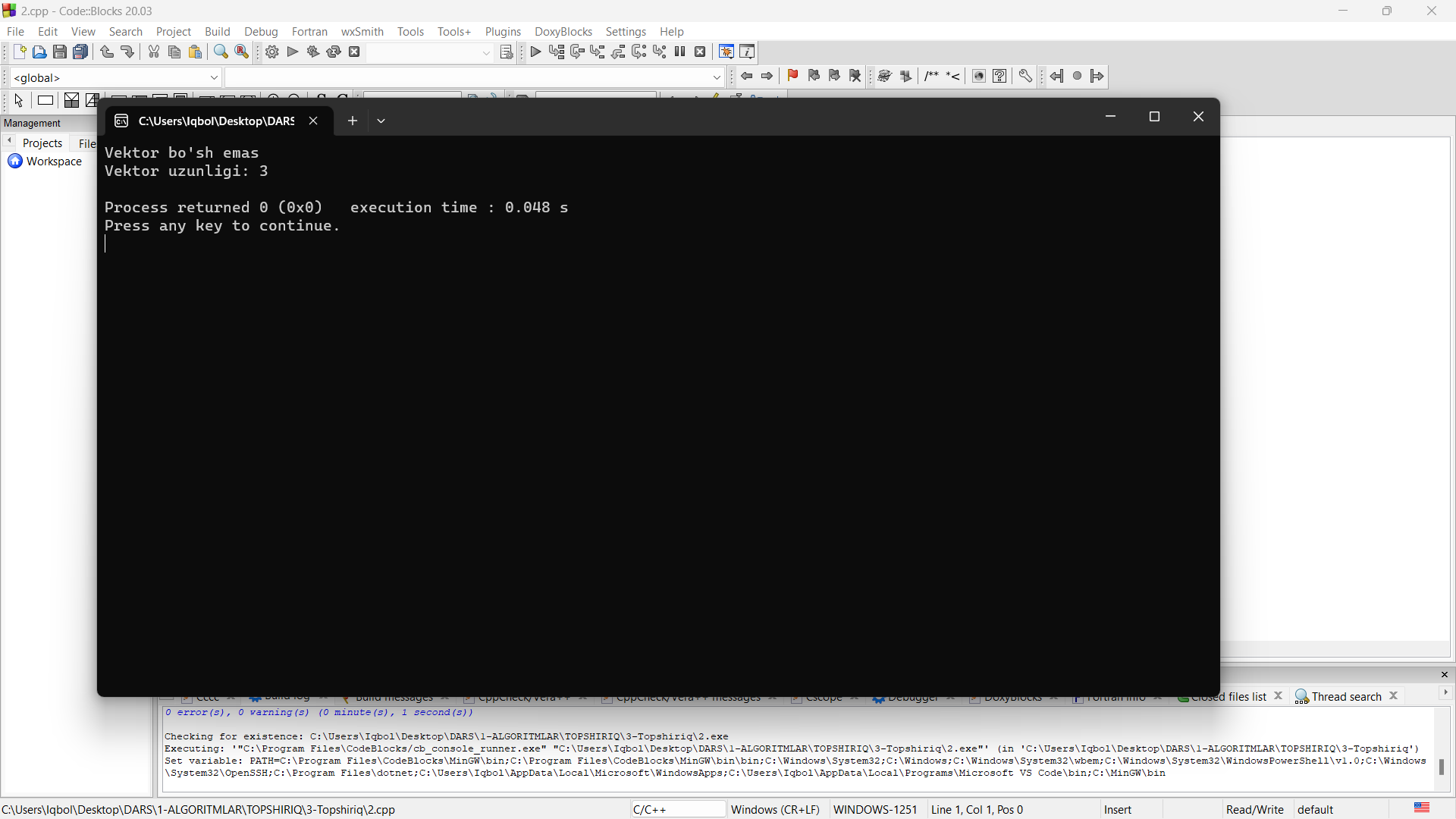
  // Vektor uzunligini aniqlash

  cout << "Vektor uzunligi: " << V.size() << endl;

  return 0;

}

//NATIJA:



4. 50 ta elemntdan iborat vektor hosil qiling va ixtiyoriy sonlar generatori bilan bir va ikki xonali sonlar bilan to’ldiring. Push\_back va pop\_back metodidan foydalaning.

#include <iostream>

#include <vector>

#include <cstdlib> // rand() funksiyasi uchun

using namespace std;

int main() {

    // 50 elementdan iborat vektor hosil qilish

    vector<int> v;

    // ixtiyoriy sonlar bilan to'ldirish

    for (int i = 0; i < 50; i++) {

        int random\_num = rand() % 90 + 10; // ixtiyoriy ikki xonali son hosil qilish

        v.push\_back(random\_num); // push\_back metodidan foydalanish

    }

    // Vektorni ekranga chiqarish

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

        cout << v[i] << " ";

    }

    cout << endl;

    // Vektordan 10 ta elementni olib tashlash

    for (int i = 0; i < 10; i++) {

        v.pop\_back(); // pop\_back metodidan foydalanish

    }

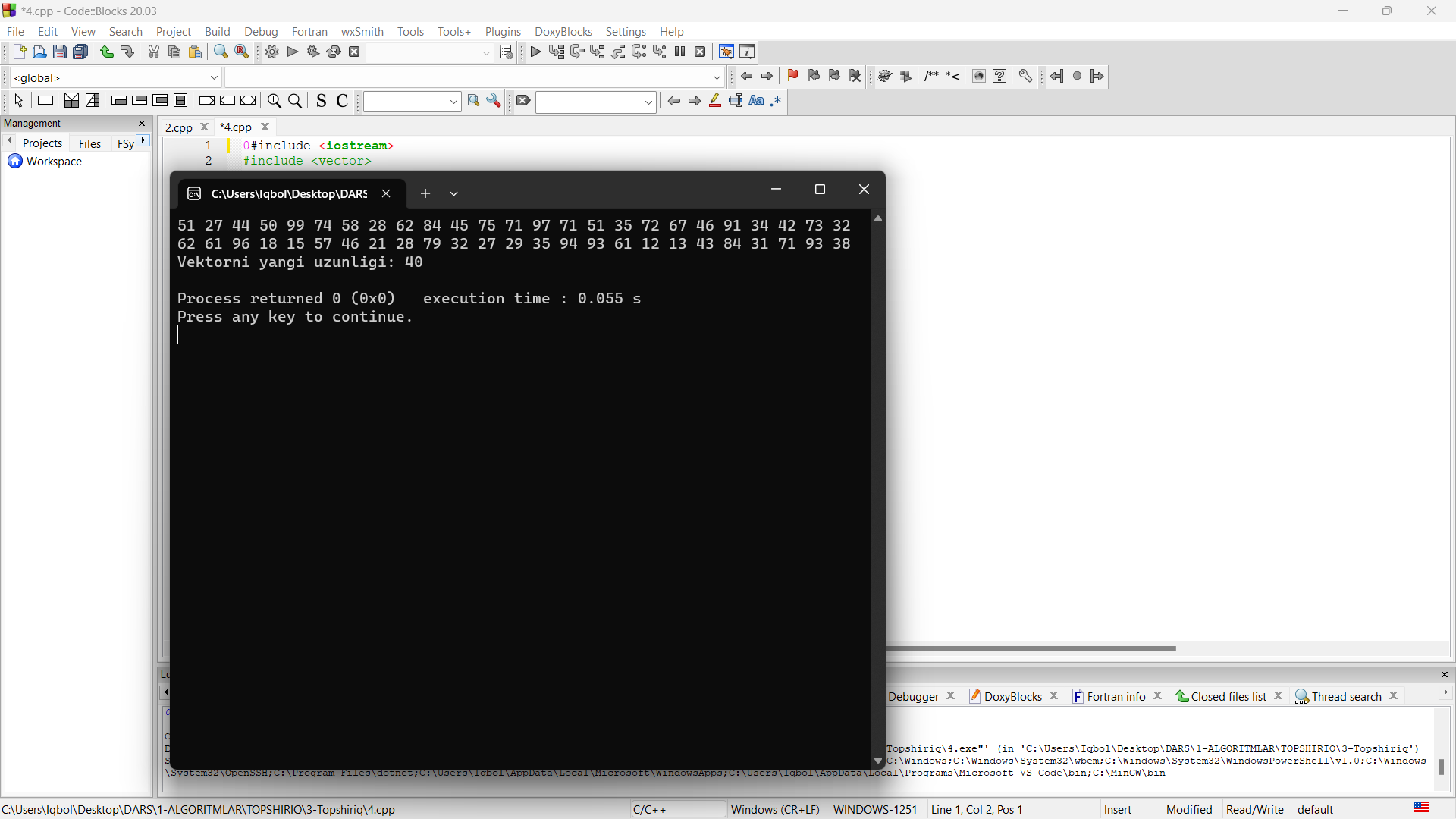
    // Yangi uzunligi chiqarish

    cout << "Vektorni yangi uzunligi: " << v.size() << endl;

    return 0;

}

//NATIJA:



6. Vektor hosil qiling insert(), begin(), end() metodlaridan foydalaning.

#include <iostream>

#include <vector>

using namespace std;

int main() {

  vector<int> my\_vector;

  my\_vector.push\_back(10);

  my\_vector.push\_back(20);

  my\_vector.push\_back(30);

  my\_vector.push\_back(40);

  for (auto it = my\_vector.begin(); it != my\_vector.end(); ++it) {

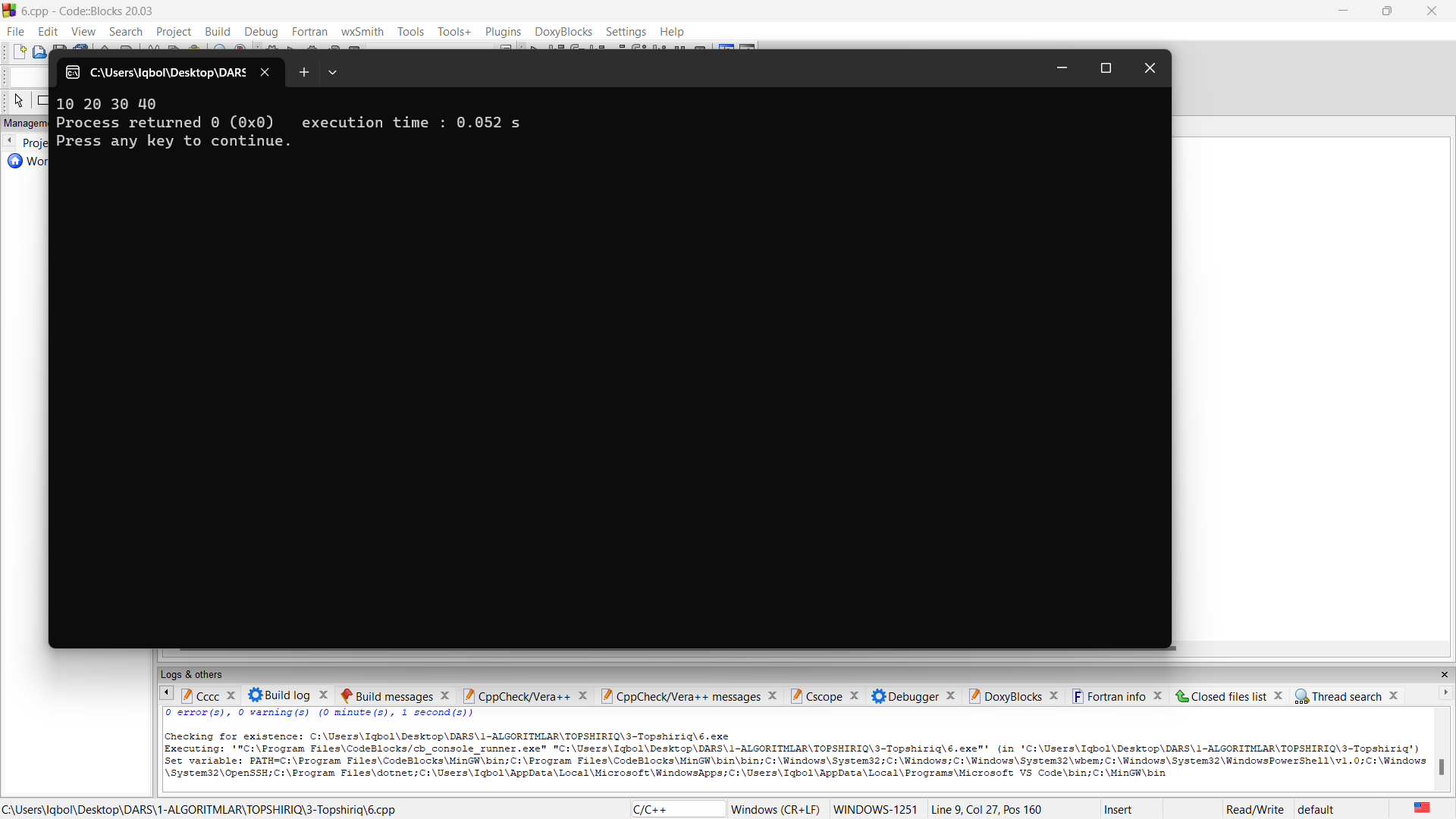
    cout << \*it << " ";

  }

  return 0;

}

//NATIJA:



8. Vektorlar vektorini hosil qiling. 3 x 3 elementdan tashkil topsin.

#include <iostream>

#include <vector>

using namespace std;

int main() {

    // 3x3 vektorlar vektorini hosil qilish

    vector<vector<int>> v(3, vector<int>(3));

    // Matritsaning qiymatlarini to'ldirish

    for (int i = 0; i < 3; i++) {

        for (int j = 0; j < 3; j++) {

            cout << " Elemetlarni kriting (" << i << ", " << j << "): ";

            cin >> v[i][j];

        }

    }

    // Matritsaning qiymatlarini chiqarish

    for (int i = 0; i < 3; i++) {

        for (int j = 0; j < 3; j++) {

            cout << v[i][j] << " ";

        }

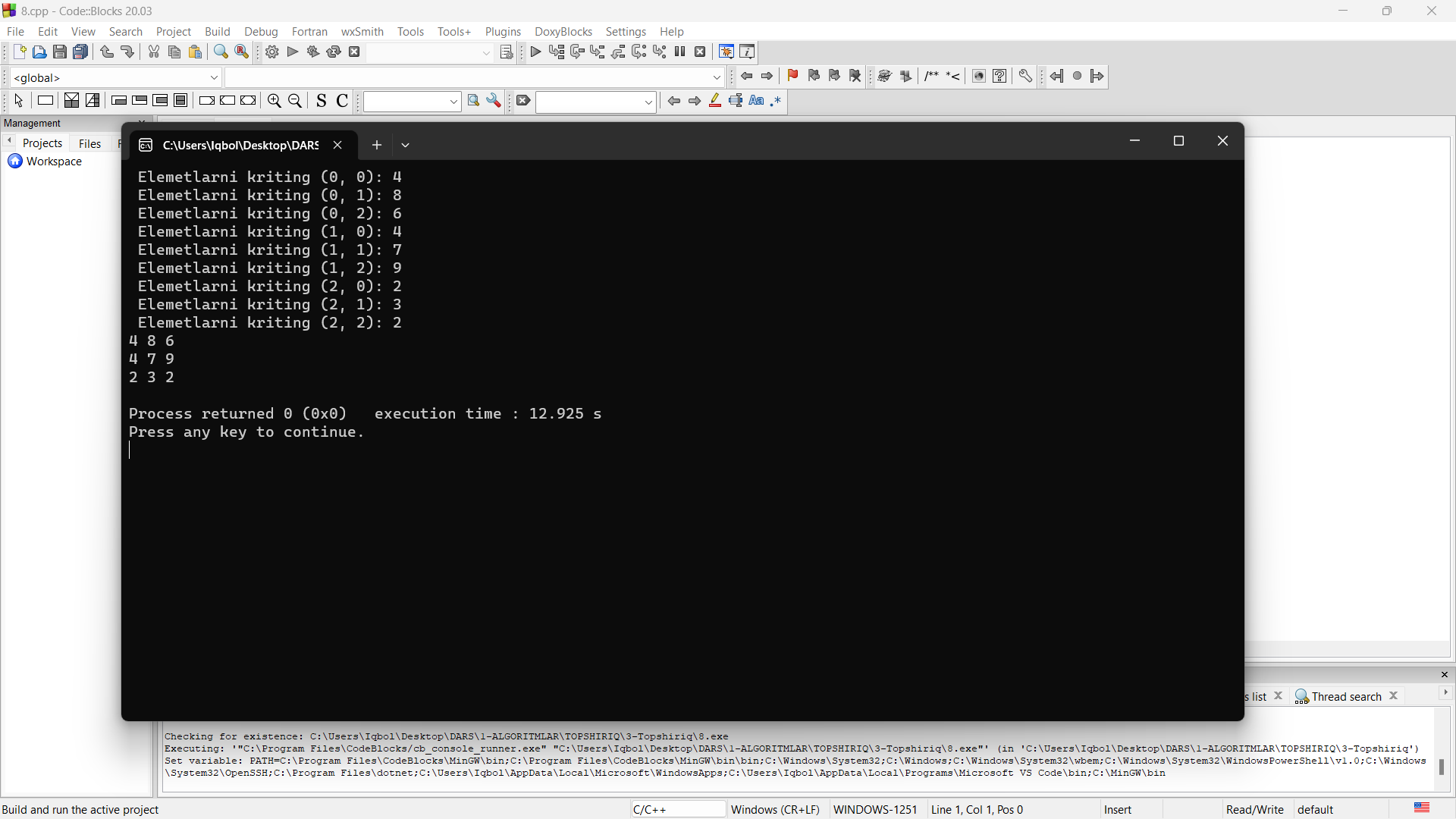
        cout << endl;

    }

    return 0;

}

//NATIJA:



10. Ro’yxat hosil qiling. Undan 2 ta elemantni o’chirib, n ta elemant qo’shing.

#include <iostream>

#include <vector>

using namespace std;

int main() {

    // Ro'yxat hosil qilish

    vector<int> V {1, 2, 3, 4, 5};

    // Ro'yxatdagi 2 ta elementni o'chirish

    V.pop\_back();

    V.pop\_back();

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

        cout <<V[i] << " ";

    }

    cout << endl;

    // Ro'yxatga n ta element qo'shish

    int n, a;

    cout << " N = ";

    cin >> n;

    for (int i = 0; i < n; i++) {

        cin >> a;

        V.push\_back(a);

    }

    // Ro'yxatdagi hamma elementlarni chiqarish

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

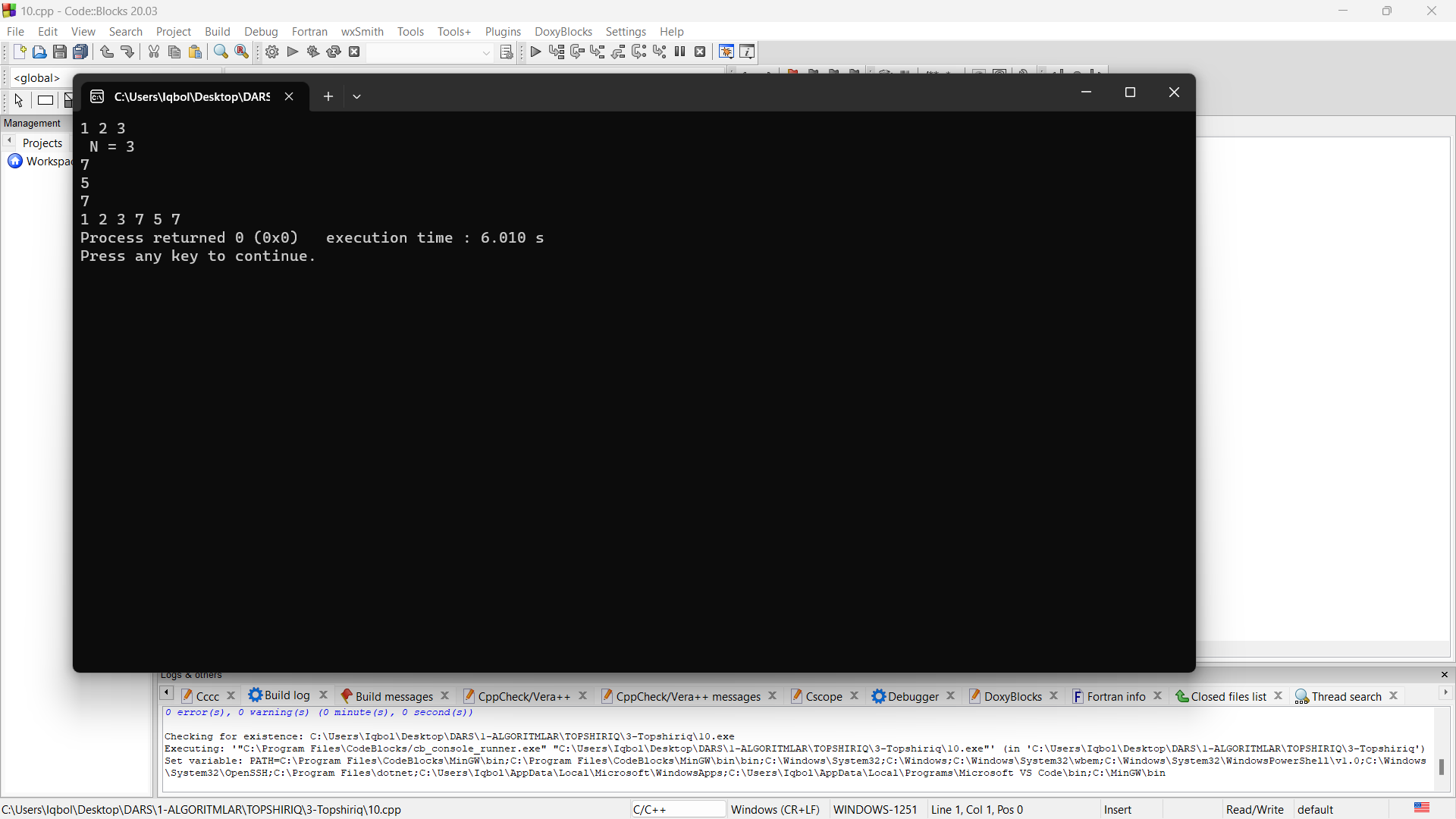
        cout << V[i] << " ";

    }

    return 0;

}

//NATIJA:



12.Ro’yxat hosil qiling birinchi va oxirgi elementini o’chiring.

#include <iostream>

#include <vector>

using namespace std;

int main()

{

    vector<int> V = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}; // ro'yxatni hosil qilamiz

    V.erase(V.begin());                              // birinchi elementni o'chirish

    V.pop\_back();                                    // oxirgi elementni o'chirish

    for (int i : V)

    {

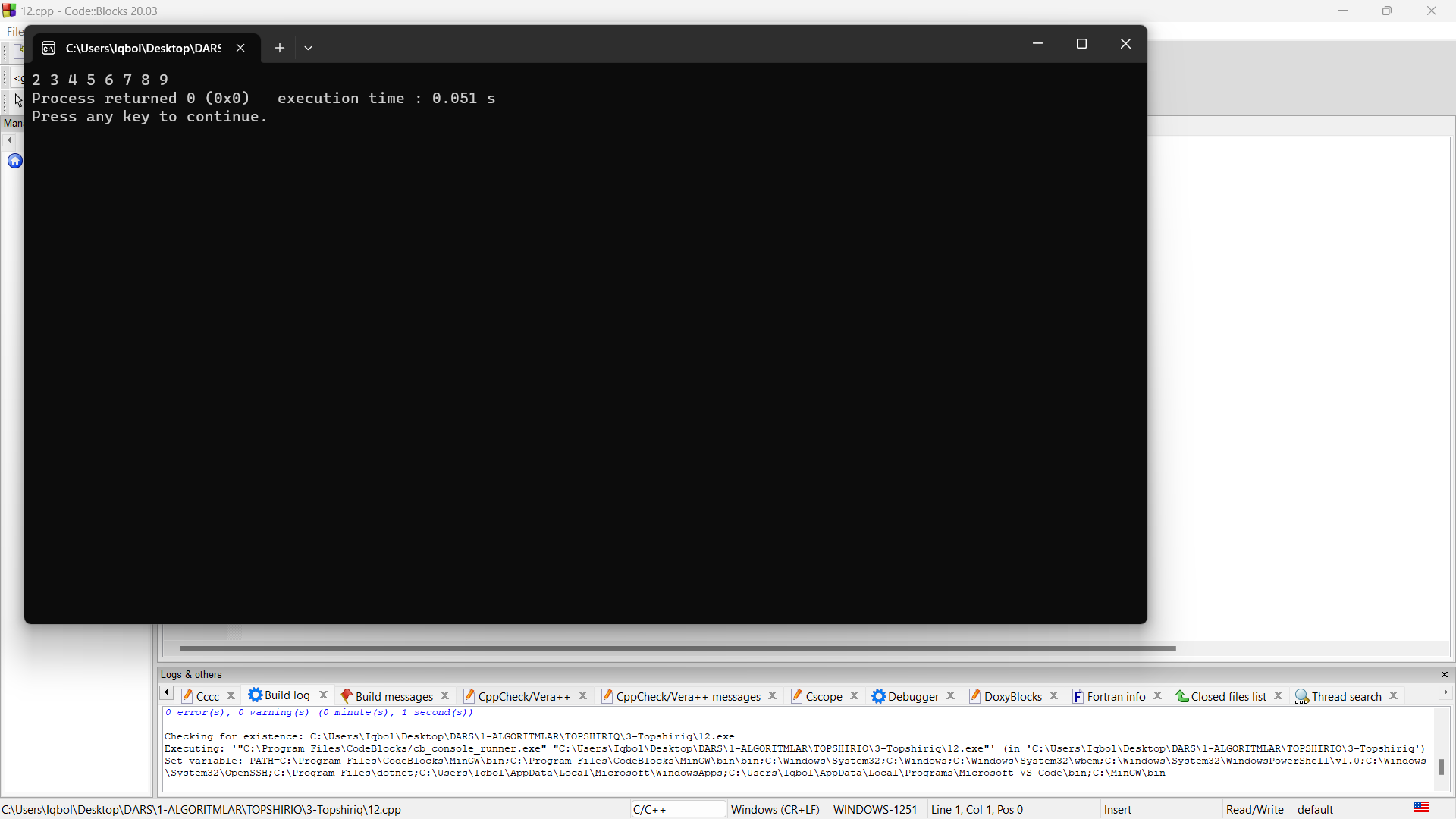
        cout << i << " ";

    }

    return 0;

}

//NATIJA:



14.Ro’yxat hosil qiling. Insert(), merge() metodlaridan foydalaning.

#include <iostream>

#include <list>

using namespace std;

int main()

{

    list<int> L1{1, 2, 3};

    list<int> L2{4, 5, 6};

    // L2 ni L1 ga birlashtiramiz qilamiz

    L1.merge(L2);

    // L1 ga yangi element qo'shamiz

    L1.insert(L1.begin(), 0);

    // L1 ni chiqaramiz

    for (int i : L1)

    {

        cout << i << " ";

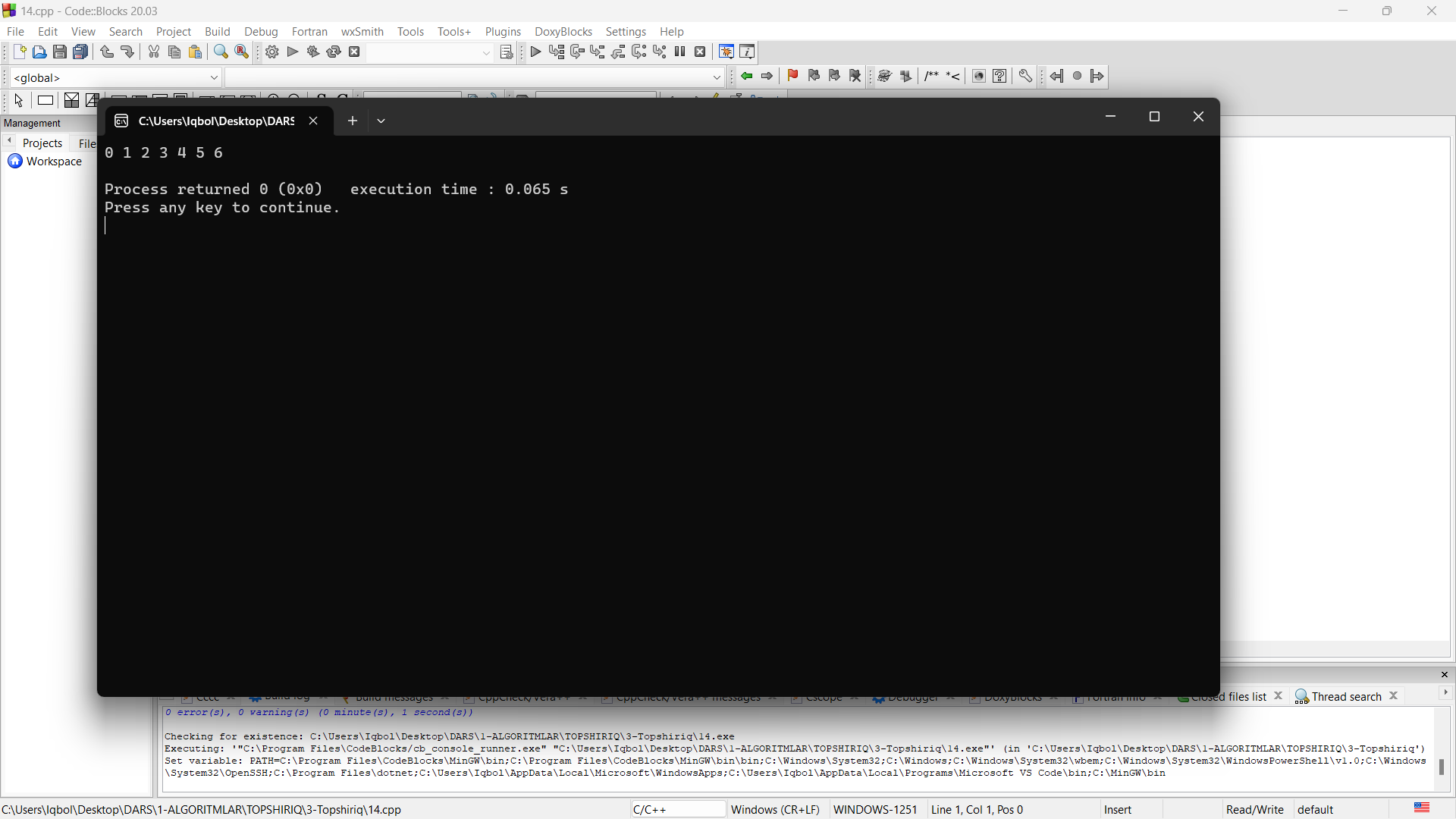
    }

    cout << endl;

    return 0;

}

//NATIJA:



16. N ta elementdan iborat ro’yxat hosil qiling. Oxirgi elementiga teng elementlar sonini toping.

#include <iostream>

#include <vector>

using namespace std;

int main()

{

    // Ro'yxatni yaratamiz va qiymatlarni kiritamiz

    vector<int> V = {2, 5, 8, 2, 7, 2, 6, 1, 0, 2};

    int c = 0;

    // Ro'yxat aylanadi, har bir element uchun teng elementlar sonini hisoblaymiz

    for (int i = 0; i < V.size(); i++)

    {

        if (V[V.size() - 1] == V[i])

        {

            c++;

        }

    }

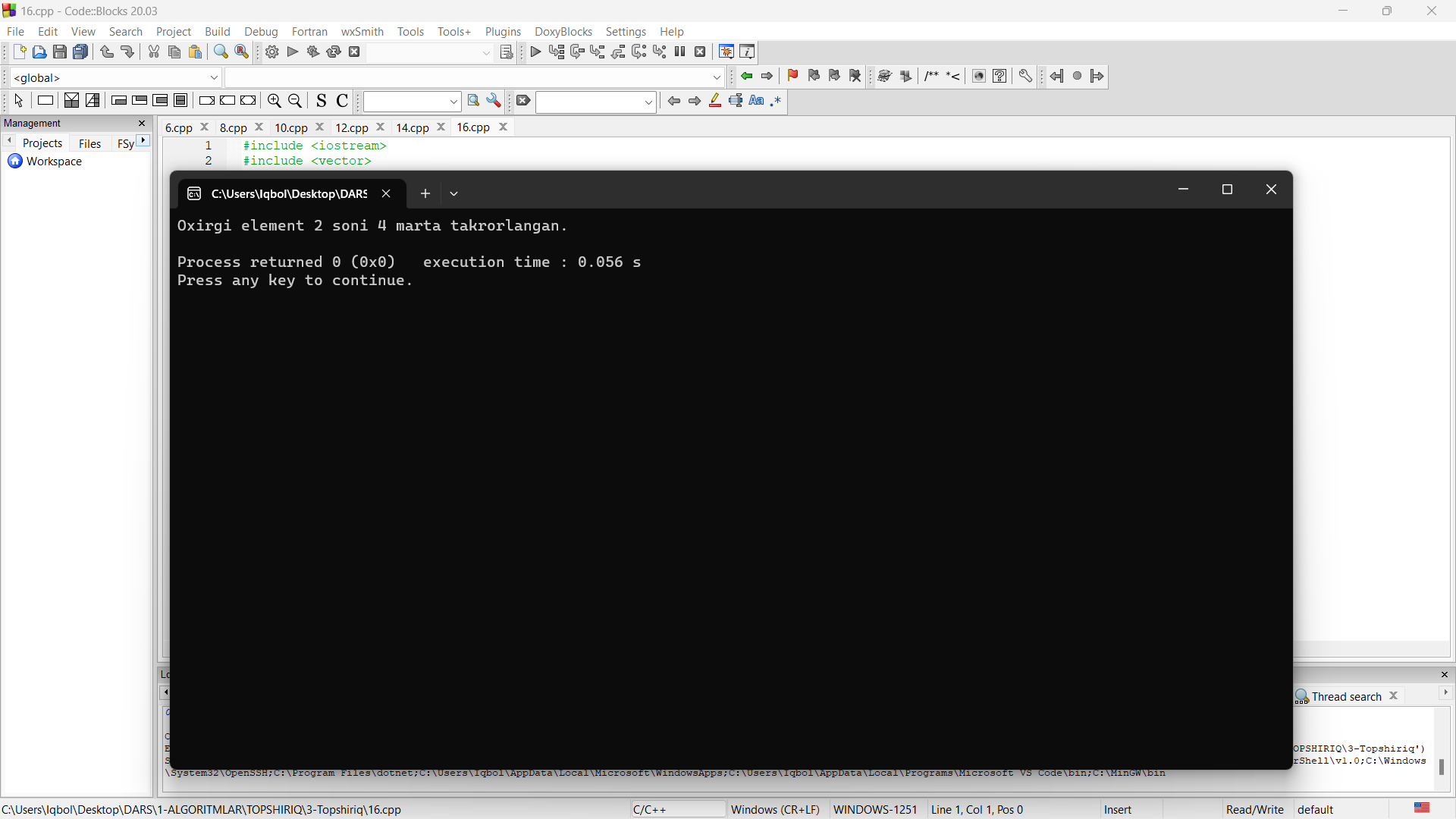
    // Oxirgi elementga teng elementlar sonini yozamiz

    cout << "Oxirgi element " << V[V.size() - 1] << " soni " << c << " marta takrorlangan." << endl;

    return 0;

}

//NATIJA:



18.N ta element iborat vektor hosil qiling. Vektorni elementlar bilan toʻldiring. Vektorga oxiridan element qoʻshing, elementni olib tashlang. Boshiga element qoʻshing.

#include <iostream>

#include <vector>

using namespace std;

int main()

{

    int n, c = 0, k = 0;

    vector<int> v;

    cout << "N ni kiriting: ";

    cin >> n;

    for (int i = 0; i < n; i++)

    {

        int x;

        cout << i + 1 << "-elementni kiriting: ";

        cin >> x;

        v.push\_back(x);

    }

    for (int i = 0; i < v.size(); i++)

    {

        if (v[i] > 0)

        {

            c++;

        }

        else if (v[i] < 0)

        {

            k++;

        }

    }

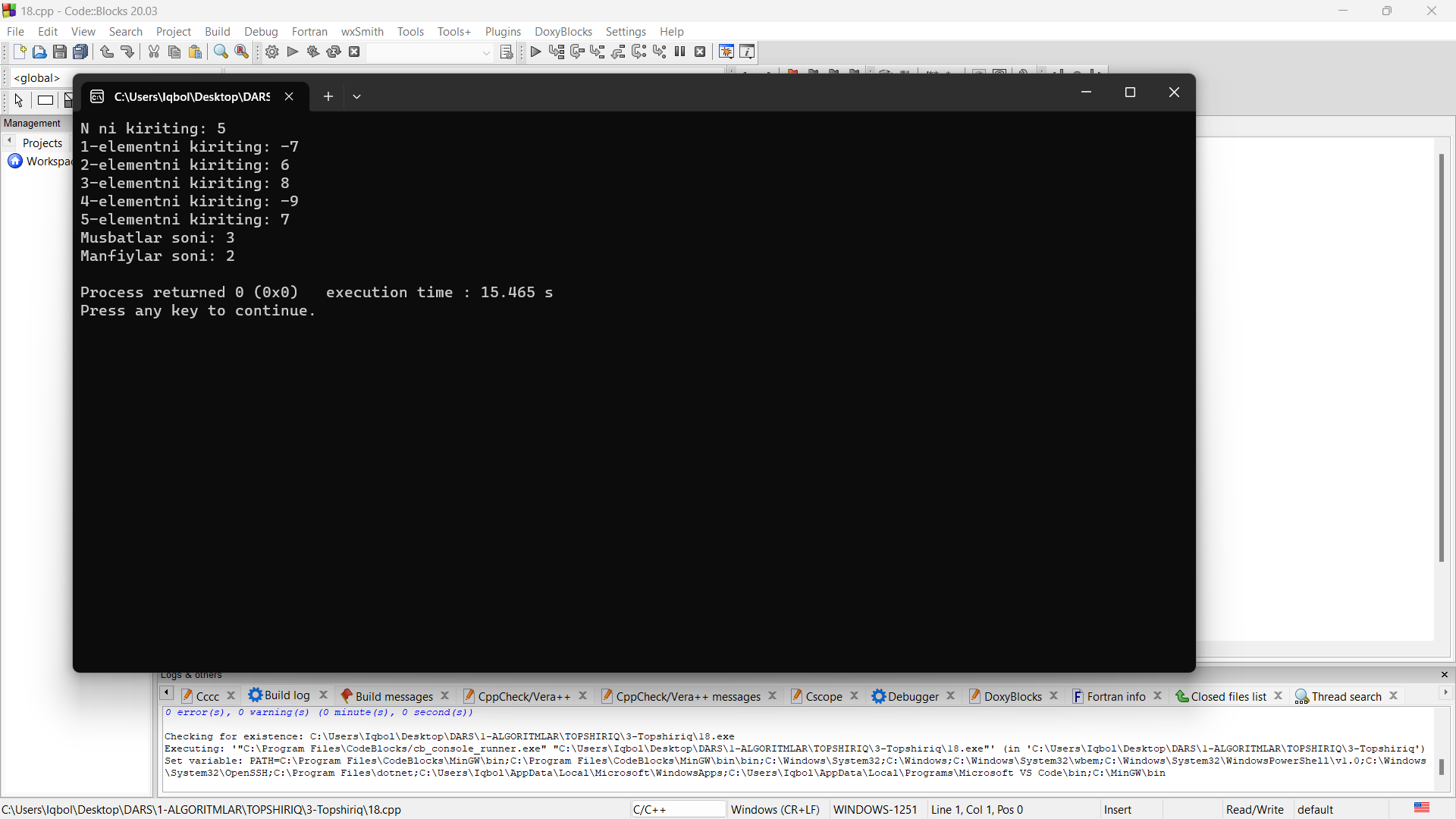
    cout << "Musbatlar soni: " << c << endl;

    cout << "Manfiylar soni: " << k << endl;

    return 0;

}

//NATIJA:



20. Birinch ro’yxat ilk n ta toq sondan, ikkinchi ro’yxat m ta ilk juft sondan iborat ro’yxat hosil qiling va ularni birlashtiring

#include <iostream>

#include <vector>

using namespace std;

int main() {

    int n, m;

    cout << " N = ";

    cin >> n;

    cout << " M = ";

    cin >> m;

    // birinchi ro'yxat

    vector<int> a(n);

    for (int i = 0; i < n; i++) {

        a[i] = 2 \* i + 1;

    }

    // ikkinchi ro'yxat

    vector<int> b(m);

    for (int i = 0; i < m; i++) {

        b[i] = 2 \* (m - i);

    }

    // birinchi ro'yxatni oxiriga ikkinchi ro'yxatni birlashtirish

    a.insert(a.end(), b.rbegin(), b.rbegin() + m);

    // chiqarish

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

        cout << a[i] << " ";

    }

    cout << endl;

    return 0;

}

//NATIJA:

