Disciplina*: Programarea structurată*  
Lucrare individuală nr. 1

*Executat:* Pricop Maxim  
*Grupa:* P-2413  
*Verificat:* Natalia Gairunova

**Partea I – Programe Liniare**

Eu sunt numărul 26 în catalog, ceea ce înseamnă că trebuie să rezolv problema 26.

**Sarcină:**

Se consideră cunoscute laturile unui dreptunghi. Sa se scrie un program in C++ care calculează aria și perimetrul acestui dreptunghi.

**Rezolvare:**

void program1() {

    float lungime, latime;

    cout << "Introdu lungimea dreptunghiului: "; cin >> lungime;

    cout << "Introdu latimea dreptunghiului: "; cin >> latime;

    if (lungime <= 0 || latime <= 0) {

        cout << "Unul din numerele introduse este mai mic sau egal ca 0. Este imposibil de a efectua calculul." << endl;

        return;

    };

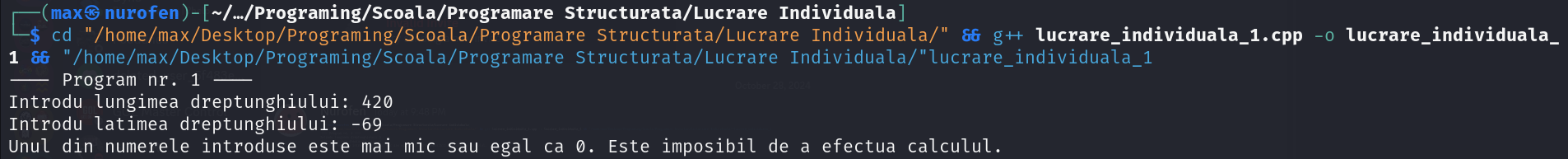
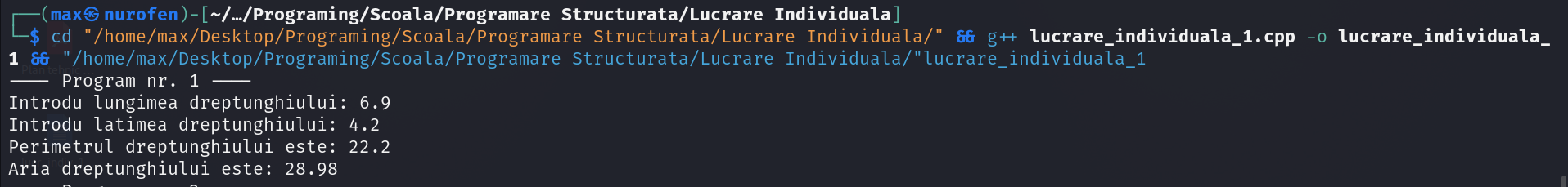
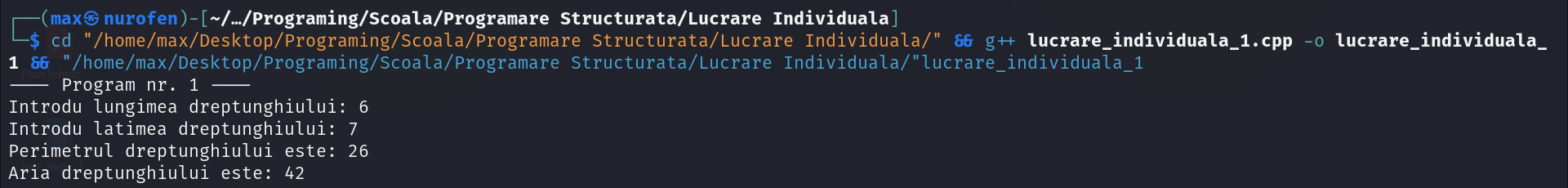
    float perimetru = 2 \* (lungime + latime);

    float aria = lungime \* latime;

    cout << "Perimetrul dreptunghiului este: " << perimetru << endl;

    cout << "Aria dreptunghiului este: " << aria << endl;

};

**Terminal:**

**Partea II – Instrucțiuni de Selecție**

Eu sunt numărul 26 în catalog, ceea ce înseamnă că trebuie să rezolv problema 9.

**Sarcină:**

Sa se alcatuiasca programul pentru functia data utilizind instructiunea IF.

F(x)=

**Rezolvare:**

void program2() {

    float x, y;

    cout << "Introdu numarul: "; cin >> x;

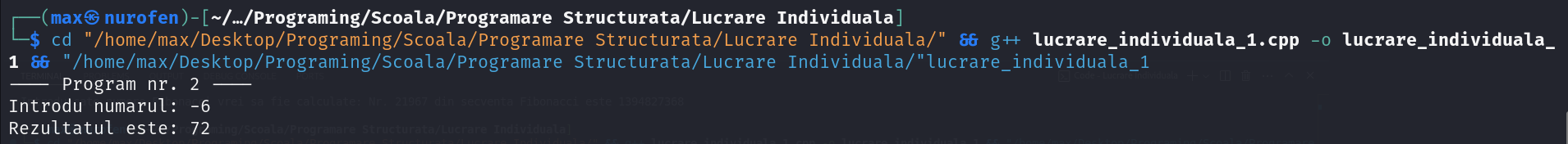
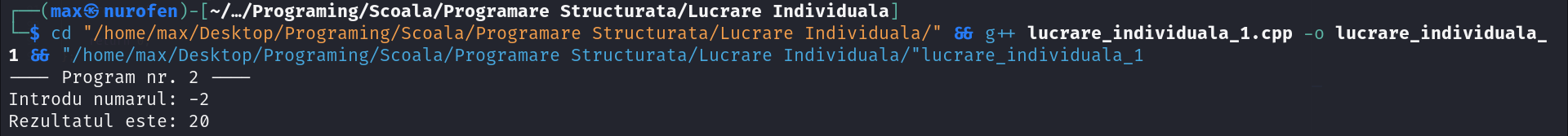
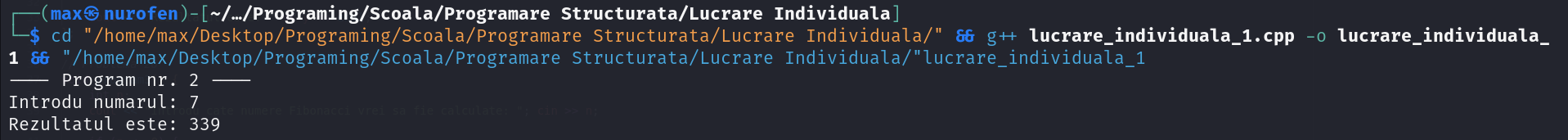
    if (x < -4) y = x + 2 \* pow(x, 2) + 6;

    else if (-4 <= x && x <= 4) y = (2 \* x \* (pow(x, 2) + 1)) / (x + 1);

    else y = 2 \* pow(x, 3) + (x - 11) - 7 \* pow(x, 2);

    cout << "Rezultatul este: " << y << endl;

};

**Terminal:**

**Partea III – Instrucțiuni Repetitive**

Eu sunt numărul 26 în catalog, ceea ce înseamnă că trebuie să rezolv problema 26.

**Sarcină:**

Sa se scrie un program in C++ care afla ultimul element n din sirul Fibonacci si il afiseaza la ecran, unde n este introdus de la tastatura.

**Rezolvare:**

void program3() {

    int n;

    cout << "Introdu cate numere Fibonacci vrei sa fie calculate: "; cin >> n;

    if (n <= 0) {

        cout << "Trebuie sa introduci un numar mai mare ca 0." << endl;

        return;

    } else if (n == 1) {

        cout << "Nr. 1 din secventa Fibonacci este 0." << endl;

        return;

    } else if (n == 2) {

        cout << "Nr. 2 din secventa Fibonacci este 1." << endl;

        return;

    };

    int i = 3, firstNumber = 0, secondNumber = 1, lastNumber;

    // Metoda 1 - For Loop

    for (i; i <= n; ++i) {

        lastNumber = firstNumber + secondNumber;

        firstNumber = secondNumber;

        secondNumber = lastNumber;

    };

    // Metoda 2 - While Loop

    // while (i <= n) {

    //     lastNumber = firstNumber + secondNumber;

    //     firstNumber = secondNumber;

    //     secondNumber = lastNumber;

    //     i++;

    // };

    // Metoda 3 - Do While Loop

    // do {

    //     lastNumber = firstNumber + secondNumber;

    //     firstNumber = secondNumber;

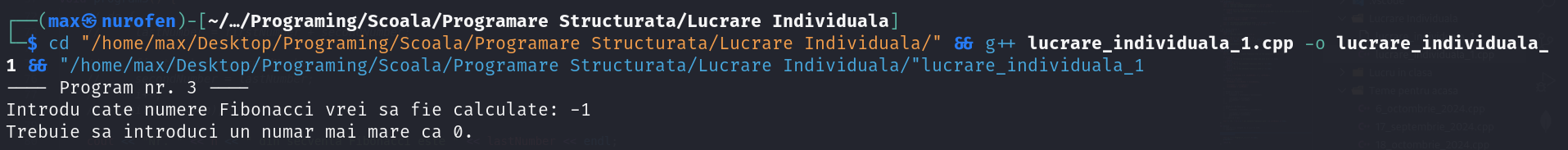
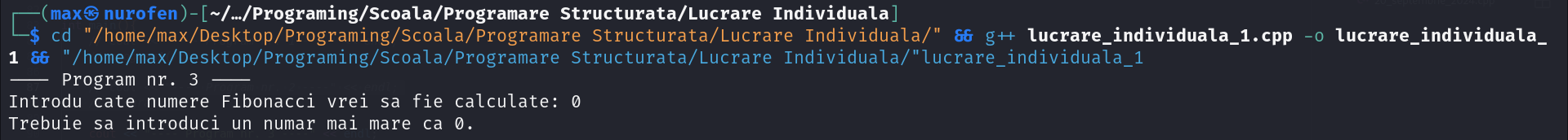
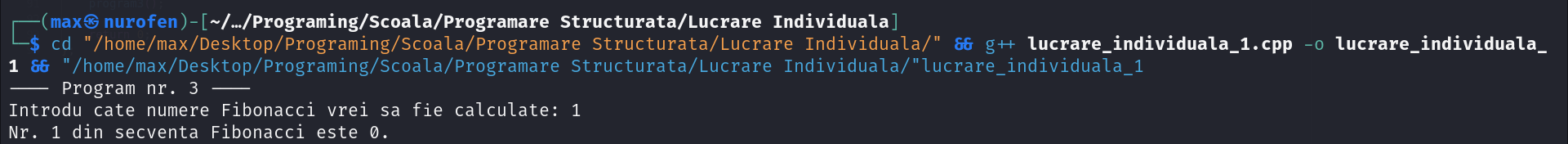
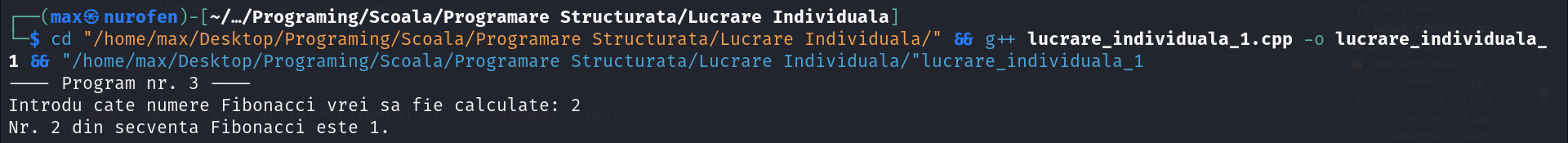
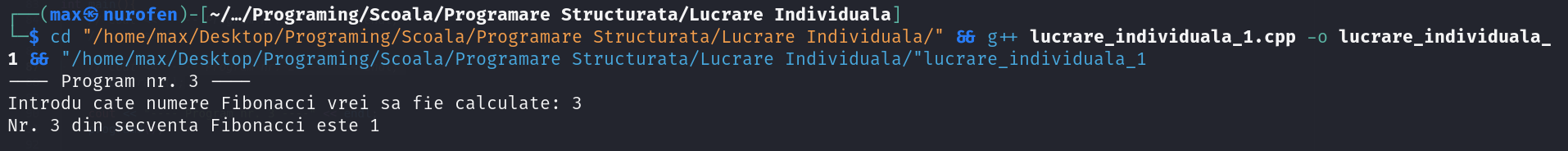
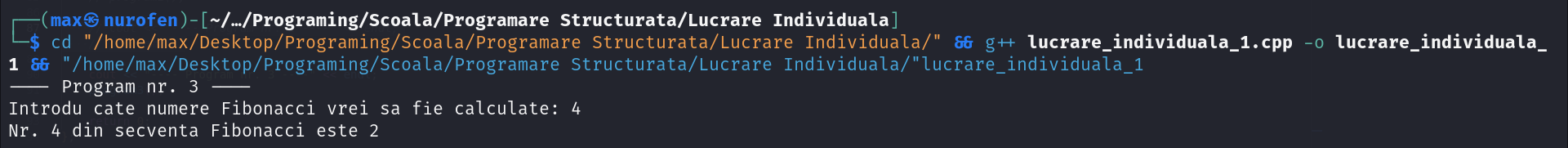
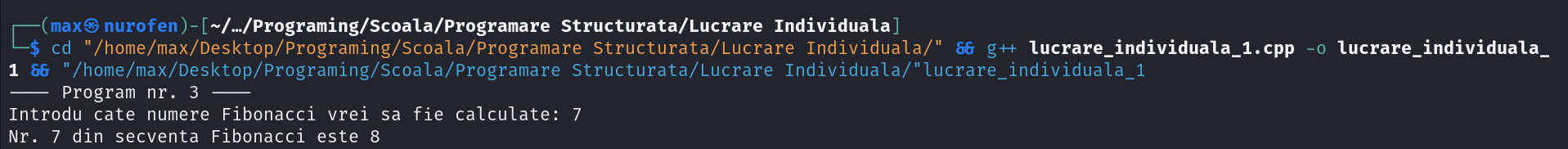
    //     secondNumber = lastNumber;

    //     i++;

    // } while (i <= n);

    cout << "Nr. " << n << " din secventa Fibonacci este " << lastNumber << endl;

};

**Terminal:**

**Concluzie**

Concluzia

**Siteuri Folosite**

W3Schools: <https://www.w3schools.com/cpp/cpp_for_loop.asp>