

Exercițiu. Să se afișeze primii 100 de termeni ai șirului (x_i) dat de relația de recurență

$$x_{i+1} = 1 - x_i x_{i-1}^3, \quad i = 1, 2, 3, \dots$$

unde

$$x_0, x_1 \in (0, 1)$$

sunt dați de utilizator, și să se determine valoarea maximă a termenilor afișați.

Indicație:

Recurențe de ordin II, principiul curgerii timpului: *trecutul se pierde, prezentul trece în trecut, viitorul devine prezent.*

Principiul valorii optime: *presupunem că prima este optimă, apoi o actualizăm.*

```
#include<iostream>
using namespace std;
void afiseaza0(double x0, double x1, int ifinal){
    double xtrec = x0, xprez = x1, xviit;
    cout << "i=0" << " x=" << xtrec << endl;
    for (int i = 1; i <= ifinal; i++){
        cout << "i=" << i << " x=" << xprez << endl;
        xviit = 1 - xprez*xtrec*xtrec*xtrec;
        xtrec = xprez;
        xprez = xviit;
    }
}
void afiseaza(double xtrec, double xprez, int ifinal){
    cout << "i=0" << " x=" << xtrec << endl;
    for (int i = 1; i <= ifinal; i++){
        cout << "i=" << i << " x=" << xprez << endl;
        double xviit = 1 - xprez*xtrec*xtrec*xtrec;
        xtrec = xprez;
        xprez = xviit;
    }
}
void afiseazaMax(double xtrec, double xprez, int ifinal){
    int imax = 0;
    double vmax = xtrec;
    cout << "i=0" << " x=" << xtrec << endl;
    for (int i = 1; i <= ifinal; i++){
        cout << "i=" << i << " x=" << xprez << endl;
        if (vmax < xprez){
            imax = i;
            vmax = xprez;
        }
        double xviit = 1 - xprez*xtrec*xtrec*xtrec;
        xtrec = xprez;
        xprez = xviit;
    }
    cout << "imax=" << imax << " vmax=" << vmax << endl;
    return;
}
int main(){
    double x0 = 0.4, x1 = 0.1;
    int ifinal = 100;
    cout.precision(12);
    afiseazaMax(x0, x1, ifinal);
    return 0;
}
```