

## PROIECT DE PROGRAME NR. 2. ORGANIZAREA CICLURILOR ÎN PROGRAM. FUNCȚII.

### I. Formularea problemei

De efectuat următoarele operații conform variantei:

1. De verificat dacă numărul nr satisface proprietatea de bază.
2. De indicat toate numerele din intervalul  $[a; b]$ , care satisfac proprietatea de bază.
3. De indicat primele  $n$  numere, care satisfac proprietatea de bază.
4. De indicat toate numerele din intervalul  $[a; b]$ , care satisfac proprietatea de bază (variante optimizată) sau care satisfac a doua proprietate.
5. De determinat cel mai mare număr, care satisface condiția indicată.

### II. Indicații metodice

Programul trebuie să fie universal, adică trebuie să se obțină rezultatul pentru orice date inițiale.

### III. Programul

```
#include <iostream>

using namespace std;

int a, b, n, n1, count = 0;

bool NumPerfect(int n) {
    int s = 1;
    for (int d = 2; d <= n / 2; d++)
        if (n % d == 0)
            s += d;
    return (n == s);
}

bool NumPrietene(int n, int s) {
    int m = 1, k = 1;
    for (int d = 2; d <= n / 2; d++) {
        if (n % d == 0)
            m += d;
    }

    for (int g = 2; g <= s / 2; g++) {
        if (s % g == 0)
            k += g;
    }

    return (s == m && n == k);
}
```

```

int main() {
    int max;

    cout << "Numar perfect = " << endl;
    cin >> n;
    cout << "Primele n numere afisate = " << endl;
    cin >> n1;

    NumPerfect(n) ? cout << n << " este perfect!" : cout << n << " nu este
perfect!" << endl;

    cout << "Indicati primul numar al intervalului " << endl;
    cin >> a;
    cout << "Indicati ultimul numar al intervalului " << endl;
    cin >> b;

    cout << "Toate numerele perfecte din interval " << endl;
    for (int i = a; i < b; i++) {
        (NumPerfect(i) && i != 1) && cout << i << "\n" << endl;
    }

    cout << "Primele " << n1 << " numere perfecte din interval \n" << endl;
    for (int i = a; i < b; i++) {
        if (NumPerfect(i) && i != 1) {
            if (count <= n1 - 1) {
                cout << i << "\n" << endl;
                count++;
            }
        }
    }

    for (int j = a; j < b; ++j) {
        for (int o = a; o < b; ++o) {
            if (j != o) {
                NumPrietene(j, o) && cout << j << " ," << o << " numere
prietene \n" << endl;
            }
        }
    }

    for (int i = INT_MAX; i < 0; i--) {
        if (NumPerfect(i) && i != 1) {
            max = i;
            break;
        }
    }
    cout << "Numarul maxim este " << max << endl;
}

```

#### IV. Rezultate

Numar perfect =

28

Primele n numere afisate =

4

28 este perfect!

Indicati primul numar al intervalului

1

Indicati ultimul numar al intervalului

9000

Toate numerele perfecte din interval

6

28

496

8128

Primele 4 numere perfecte din interval

6

28

496

8128

Numerele prietene din interval

220 ,284 / 284 ,220

1184 ,1210 / 1210 ,1184

2620 ,2924 / 2924 ,2620

5020 ,5564 / 5564 ,5020

6232 ,6368 / 6368 ,6232

Numarul maxim este

4201419