

Suspectul I

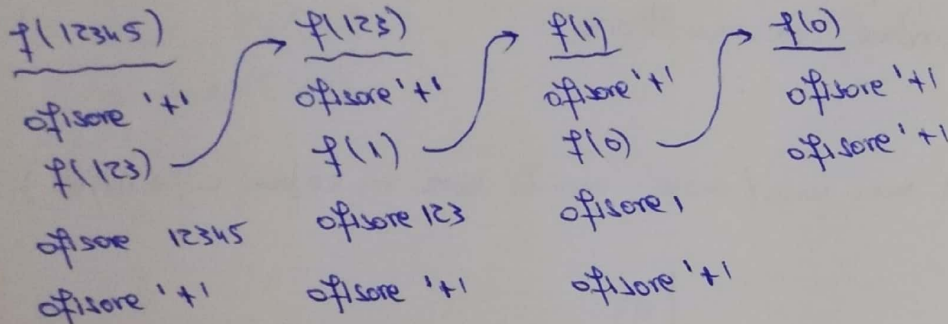
$$1. \quad ! (x < 2001) \text{ and } (x \leq 2002) \text{ or } (x \geq 2019) \text{ and } ! (x > 2020)$$

\Leftrightarrow

$$(x \geq 2001) \text{ and } (x \leq 2002) \text{ or } (x \geq 2019) \text{ and } (x \leq 2020)$$

d)

2.



$$\Rightarrow f(0) \rightarrow \text{ofisore}++$$

$$\Rightarrow f(1) \rightarrow \text{ofisore}+++1+$$

$$\Rightarrow f(123) \rightarrow \text{ofisore}++++1+123+$$

$$\Rightarrow f(12345) \rightarrow \text{ofisore}+++++1+123+12345+$$

a)

3. Se generează mulțimile de 6 numere câte 4. Condiția ca o mulțime să fie validă este ca elementele mulțimii să fie în ordine strict crescătoare.

Se observă că singura variantă de indexare a acestor aspect este a doua

b)

4. Pentru a determina gradul intercom de unii nodi conform tuturor elementelor de pe coloana corespunzătoare.

c)

5. Indiferent de numărul de moduri pe care îl obținem unei componente care, răspunsul va fi 15.

↳ general, pentru a obține x componente care să aibă la dispoziție m moduri, numărul minim necesar de mulți (componente care să obținute va fi cel) va fi egal cu $m-x$

c)

Subiectul II

1. Algoritmul descrie în pseudocod formarea unui număr din cifre impare de la m , în ordine descrescătoare

a) 7533

b) 20 88 (cel mai mic/mare număr care nu conține cifre impare)

c)

```
#include <iostream>
using namespace std;
int x, m, p, cm, em;
int main()
{
    cm >= m;
    x = 1; m = 0; p = 1;
    while (x < 10)
    {
        em = m;
        while (em != 0)
        {
            c = em % 10; em /= 10;
            if (c == x)
            {
                m = c * p + m; p *= 10;
            }
        }
        x += 2;
    }
    cout << m;
    return 0;
}
```

```
d)
citeste m
m ← 0; p ← 1
pentru x ← 1, 10, 2
do
    cm ← m
    cât timp cm ≠ 0 execută
        c ← cm % 10; cm ← [cm / 10]
        dacă c = x atunci
            m ← c * p + m; p ← p * 10
    sfârșit
sfârșit m
```

2. struct biblio {

unsigned short numor;

struct {

char titre[20];

char auteur[20];

} carte[100];

} e;

3.

ou

resu