

Descrierea soluției –Problema Expoziție

Soluția 1-

Deoarece fiecare desen trebuie să apară de cel puțin k ori, ne vom asigura de acest lucru afișând fiecare desen de exact k ori, au fost ocupate astfel $k*d$ scânduri, au mai rămas libere $r=n-k*d$, fiecare desen mai apare pe lângă cele k apariții de $k_1, k_2, \dots, k_d$ ori. Dacă $r=0$ numărul de aranjări este 1. Dacă $r<0$ numărul de aranjări este 0. În celelalte cazuri considerăm ecuația:

$$k_1+k_2+k_3+\dots+k_d=r, 0 \leq k_i \leq r \quad (1)$$

Problema se reduce la a determina numărul de soluții ale ecuației (1), acest număr este egal cu

$C_{r+d-1}^r = C_{r+d-1}^{d-1}$. Pentru demonstrație se reprezintă soluția ecuației (1) ca o secvență binară, formată din k_1 elemente 0, urmate de un element 1, urmat de k_2 elemente 0, urmate de un element 1, ș.a.m.d., se adaugă în final k_d elemente 0. În secvența construită sunt r elemente 0 ($k_1+k_2+k_3+\dots+k_d=r$), numărul total de elemente este $n+r-1$. Numărul posibilităților de a aranja cele r elemente 0 este C_{r+d-1}^r .

Pentru calculul combinațiilor se utilizează triunghiul lui Pascal care se bazează pe următoarea formulă de recurență:

$$C_m^0 = 1, C_m^m = 1$$

$$C_m^n = C_{m-1}^n + C_{m-1}^{n-1}$$

Fie $m=r+d-1$, trebuie să calculăm C_m^r

În triunghiul lui Pascal vom utiliza doar 2 linii, linia curentă (L_c) și linia precedentă (L_p), deoarece rezultatul poate fi un număr foarte mare, vom implementa operațiile cu numere mari.

(exp_comb.cpp)

Soluția 2

Problema se poate rezolva și prin metoda programării dinamice. Se elimină din n cele $k*d$ scânduri pe care se afișează câte k desene de fiecare tip. Rămâne să se afișeze planșele pe $r=n-k*d$ scânduri. Dacă $r=0$ numărul de moduri de aranjare este 1. Dacă $r<0$ numărul este 0. În celelalte cazuri considerăm $a[i][j]$ =numărul modurilor de aranjare dacă avem i scânduri și j desene originale; deoarece ordinea desenelor nu contează le afișăm în ordinea crescătoare a numărului în scris pe acestea.

Este ușor de dedus că:

$a[1][j]=j$, pentru $j=1, d$ (avem o scândură și j desene)

$a[i][1]=1$, pentru $i=1, n$ (avem i scânduri și 1 desen)

$a[i][j]=a[i-1][j]+a[i][j-1]$, $i=2,n$ și $j=2,d$ ($a[i][j-1]$ reprezintă toate posibilitățile de a afișa $j-1$ desene pe i scânduri, trebuie adăugate toate posibilitățile de afișare care includ desenul j acest număr este $a[i-1][j]$, la cele $i-1$ scânduri adăugăm o scândură, pe aceasta se poate afișa doar desenul j , soluțiile fără j fiind deja numărate). Soluția se obține în $a[r][d]$.

Este suficient să memorăm doar două coloane, coloana curentă și coloana precedentă, deoarece rezultatul poate fi un număr foarte mare, vom implementa operațiile cu numere mari.

(expo_din.cpp)