



Concursul INTERJUDE- ȚEAN de programare INFO-OLTENIA 2005

Vă prezentăm în continuare soluțiile celor zece probleme propuse spre rezolvarea la ediția din acest an a concursului interjudețean de programare Info-Oltenia.

Clasa a IX-a

P050215: Șir

Construim o matrice C ale cărei elemente c_{ij} indică de câte ori apare cifra $i + 1$ pe poziția j .

Parcurgem apoi pe coloane această matrice și, dacă identificăm o valoare c_{ij} care nu este multiplu al numărului k , atunci putem trage concluzia că cifra $i + 1$ se află pe poziția j în șirul care trebuie determinat.

După terminarea acestei parcurgeri cunoaștem care sunt elementele șirului căutat.

P050216: Biți

Pentru început vom păstra în memorie primul șir de numere.

În continuare, vom citi câte un număr care face parte din cel de-al doilea șir.

Pentru fiecare al i -lea element citit, vom extrage din cel de-al i -lea element al primului șir, biții corespunzători pozițiilor din numărul citit care au valoarea 0. Acești biți vor fi păstrați într-un al treilea șir.

Folosind biții extrași vom construi numere, astfel: alegem câte opt

biți și construim un număr astfel încât bitul de pe poziția cea mai mică în cadrul grupului său (poziția 0) se va afla pe prima poziție a reprezentării binare a numărului obținut, citit de la stânga la dreapta, deci bitul de pe a șaptea poziție.

Clasa a X-a

P050217: Soldați

Această problemă se reduce la foarte cunoscuta problemă de determinarea a numărului de ordine al unei permutări date.

P050218: Frații

Această problemă poate fi rezolvată folosind metoda programării dinamice.

Vom nota cu a_{ij} valoarea maximă care poate fi obținută folosind doar fracțiile $i, i + 1, \dots, j$.

O astfel de valoare poate fi calculată considerând că acest grup de fracții este împărțit în alte două grupuri formate din fracțiile $i, i + 1, \dots, k$, respectiv $k + 1, k + 2, \dots, j$; fiecare dintre cele două grupuri de fracții va fi inclus într-o pereche de paranteze,

iar între grupuri vom adăuga fie operatorul $+$, fie operatorul $*$. Dintre toate variantele posibile pentru k , o vom alege pe cea care duce la obținerea celei mai mari valori.

Formula recurentă de calcul pentru a_{ij} este:

$$a_{ij} = \begin{cases} f_i, & \text{dacă } i = j \\ \max \{ \max (a_{ik} + a_{k+1,j}, a_{ik} \cdot a_{k+1,j}) \} & i \leq k < j \end{cases}$$

unde prin f_i a fost notată cea de-a i -a fracție.

Rezultatul va fi dat de valoarea $a_{1,N}$, iar ordinul de complexitate al întregului algoritm este $O(N^3)$.

Clasele a XI-a și a XII-a

P050219: Ecuație

Pe baza relației $A \times X = B$, aplicând determinantul, obținem $\det A \cdot \det X = \det B$, deci valoarea căutată este dată de raportul determinanților celor două matrice date.

Pentru a calcula determinantul unei matrice date poate fi utilizată metoda lui Gauss care duce la obținerea unei matrice care are același



determinant, dar conține doar elemente nule sub diagonala principală.

Determinantul unei astfel de matrice este evident dat de produsul elementelor de pe diagonala principală.

P050220: Livada

Problema se reduce la determinarea arborelui parțial minim al unui graf orientat, precum și determinarea celui de-al doilea arbore parțial minim.

Pentru rezolvarea primei cerințe a problemei, există algoritmi foarte cunoscuți (*Prim*, *Kruskal* etc.).

Al doilea arbore parțial poate fi obținut pe baza primului prin eliminarea unei muchii și adăugarea alteia, dintre cele rămase în afara arborelui minim.

Pentru fiecare muchie (i, j) din arborele parțial minim, vom determina costul maxim M al unei muchii care se află în primul arbore parțial minim pe drumul care unește nodurile i și j .

În final vom alege cea valoare pentru care diferența dintre costul său și valoarea M corespunzătoare este minimă.

Muchia respectivă va face parte din noul arbore, în timp ce muchia cu costul M care făcea parte din arborele inițial va fi eliminată.

Echipe (incepători)

P050221: Tablou

Datorită faptului că operația dată este asociativă, matricea A^K poate fi determinată prin diverse grupări ale factorilor.

O posibilă soluție se bazează pe recurență:

$$A^K = \begin{cases} A, K=1 \\ A^{K-1} \cdot A, K \neq 1 \end{cases}$$

Datorită faptului că, în total, se determină K matrice, ordinul de complexitate al algoritmului propus este $O(N^3 \cdot K)$.

O soluție mai performantă constă în determinarea a numai $O(\log K)$ matrice, deci ordinul de complexitate al noului algoritm ar deveni $O(N^3 \cdot \log K)$.

Relația de recurență utilizată în acest caz este:

$$A^K = \begin{cases} A, K=1 \\ A^{\frac{K}{2}} \cdot A^{\frac{K}{2}}, K:2 \\ A^{K-1} \cdot A, K \text{ Ț2} \end{cases}$$

Pentru a determina elementele matricei finale modulo 9901, se folosesc două proprietăți ale aritmeticii modulo 9901:

- $(A + B) \bmod 9901 = ((A \bmod 9901) + (B \bmod 9901)) \bmod 9901$
- $(A \cdot B) \bmod 9901 = ((A \bmod 9901) \cdot (B \bmod 9901)) \bmod 9901$

Pe baza acestor formule se poate observa faptul că se pot calcula și elementele matricelor intermediare tot modulo 9901, fără a afecta rezultatul final.

Se putea evita astfel implementarea operațiilor cu numere mari pentru rezolvarea problemei.

P050222: Joc

Se determină punctul care are coordonata orizontală minimă și se efectuează o translație astfel încât punctul respectiv să ajungă în originea sistemului de axe de coordonate.

După efectuarea acestei translații, toate punctele se vor afla în cadranul I sau în cadranul IV.

Se construiește în continuare un vector ale cărui elemente au câte două câmpuri.

Primul câmp reprezintă tangenta unghiului determinat de axa Ox cu dreapta care trece prin origine și prin punctul păstrat în al doilea câmp.

În continuare se ordonează crescător vectorul, în funcție de valorile primului câmp al elementelor.

În final se vor afișa valorile celui de-al doilea câmp al elementelor vectorului sortat.

Echipe (avansați)

P050223: Party

Rezolvarea acestei probleme se bazează pe metoda programării dinamice.

Vom nota prin a_i produsul maxim care poate fi obținut dacă

președintele invită la petrecere doar subalterni ai angajatului i , fără a-l invita și pe acesta și cu b_i produsul maxim care poate fi obținut dacă președintele invită la petrecere doar subalterni ai angajatului i , dar și pe acesta.

Astfel, răspunsul corect va fi reprezentat întotdeauna de valoarea b_1 .

Vom determina aceste perechi de valori pornind de la frunzele arborilor și urcând spre rădăcină, deoarece avem nevoie de aceste valori pentru toți subalternii unui angajat.

Calculul celor două valori se face pornind de la observația că dacă un angajat este prezent la petrecere, nici un subaltern direct al său nu poate fi prezent, dar dacă el nu este prezent, atunci unii subalterni pot fi prezenți (este posibil să fie prezenți toți sau să nu fie prezent nici unul).

Formulele de calcul sunt:

$$a_i = \prod_{j \text{ subaltern al lui } i} \max(a_j, b_j)$$

$$b_i = \prod_{j \text{ subaltern al lui } i} a_j$$

Folosind aceste formule și parcurgând arborele, putem calcula valorile corespunzătoare tuturor angajaților.

Pentru a evita folosirea numerelor mari, putem folosi logaritmi pentru a transforma produsele în sume:

$$\ln a_i = \sum_{j \text{ subaltern al lui } i} \ln \max(a_j, b_j)$$

$$\ln b_i = \sum_{j \text{ subaltern al lui } i} \ln a_j$$

P050224: Gali

Satele galilor pot fi privite ca fiind nodurile unui graf neorientat, iar cărările pot fi considerate a fi muchiile grafului.

În aceste condiții problema se reduce la determinarea tuturor punctelor critice ale acestui graf.

Vom utiliza algoritmul de determinare a nodurilor critice și, în momentul detectării unui astfel de nod, vom marca acest nod ca fiind critic.

În final, vom număra nodurile critice și vom scrie numerele de ordine ale acestora.