

## Greedy – 4 probleme

1. Se dau  $n$  cuburi cu laturile diferite două câte două. Fiecare cub are culoarea albă sau neagră. Să se construiască un turn cu număr maxim (sau de înălțime maximă) de cuburi în care un cub nu poate fi așezat pe un cub de aceeași culoare sau cu latură mai mică decât a sa. Justificați corectitudinea metodei propuse. Dacă lungimile laturilor cuburilor nu erau diferite mai era valabilă metoda?
2. Se dau  $n, k$  două numere naturale. Să se construiască o matrice pătratică  $n \times n$  care verifică simultan condițiile:
  - conține numerele de la 1 la  $n^2$  o singură dată
  - pe fiecare linie numerele sunt ordonate crescător de la stânga la dreapta
  - suma elementelor pe coloana  $k$  este minimă
3. Se dau mulțimile de numere întregi nenule  $A = \{a_1, a_2, \dots, a_m\}$  și  $B = \{b_1, b_2, \dots, b_n\}$ , unde  $m \leq n$ . Să se determine  $B' = \{x_1, x_2, \dots, x_m\}$  o submulțime a lui  $B$  astfel încât suma  $a_1 x_1 + a_2 x_2 + \dots + a_m x_m$  să fie maximă. **Exemplu:** Pentru  $A = \{2, 4, 3\}$  și  $B = \{5, -3, 8, -1, 2\}$  suma maximă este 51 și se obține pentru  $B' = \{2, 8, 5\}$
4. Interclasarea optimă a  $n$  șiruri ordonate – Se dau lungimile a  $n$  șiruri ordonate  $L_1, L_2, \dots, L_n$ . Dorim să obținem un șir ordonat crescător care conține toate elementele celor  $n$  șiruri inițiale, interclasând succesiv perechi de șiruri. Știind că interclasarea a două șiruri de lungimi  $A$  respectiv  $B$  necesită  $A+B$  deplasări, să se determine o ordine în care trebuie să se realizeze interclasările astfel încât numărul total de deplasări să fie minim.
5. **Problema partiționării intervalelor** – Se consideră  $n$  cursuri, pentru care se cunosc ora de început și ora de sfârșit. Determinați numărul minim de săli necesare pentru a putea programa aceste cursuri în aceeași zi și afișați o astfel de programare. Exemplu: pentru  $n=4$  cursuri, care trebuie să se desfășoare în intervalele:  $[10, 14]$ ,  $[13, 16]$ ,  $[11, 12]$ , respectiv  $[15, 18]$ , sunt necesare 2 săli, o programare fiind: cursurile 1 și 4 în prima sală, cursurile 2, 3 în cea de a doua sală.
6. Dat un arbore, să se determine o mulțime de vârfuri neadiacente de cardinal maxim (o submulțime independentă maximă a mulțimii vârfurilor). Pentru un graf oarecare mai este valabilă metoda?
7. La inaugurarea unui restaurant sunt prezente mai multe persoane. Clienții își aleg din meniul pus la dispoziție câte o specialitate. Deocamdată însă restaurantul a angajat un singur bucătar care pregătește mâncărurile una după alta, deci clienții nu pot fi serviți decât pe rând. Presupunând că bucătarul se apucă de lucru după ce s-au strâns toate comenzile și timpul necesar pregătirii fiecărei comenzi este cunoscut, stabiliți în ce ordine trebuie să pregătească specialitățile astfel încât timpul mediu de așteptare a clienților să fie minim.

**Exemplu:** pentru 5 clienți numerotați de la 1 la 5 cu timpii necesari pregătirii comenzilor, în ordine, pentru cele 5 persoane, 30, 40, 20, 25, 60, ordinea de servire optimă este 3, 4, 1, 2, 5, timpul mediu de așteptare fiind 86.

8. Spunem că un număr este frumos dacă are ca factori primi doar 2, 3, și 5. Dat un număr natural  $n$ , să se afișeze primele  $n$  numere frumoase ( $O(n)$ ).
9. Considerăm următorul joc pentru două persoane. Tabla de joc este o secvență de numere întregi pozitive, iar cei doi jucători mută alternativ. Când un jucător mută, el selectează un număr ori de la stânga ori de la dreapta secvenței. Numărul selectat este șters de pe tablă. Jocul se termină când toate numerele au fost selectate. Primul jucător câștigă dacă suma numerelor pe care le-a selectat este cel puțin egală cu suma selectată de al doilea jucător. Al doilea jucător joacă cât de bine poate. Primul jucător începe jocul. Dacă pe tablă se află la început un număr par de elemente, atunci primul jucător are o strategie sigură de câștig. Să se scrie un program astfel încât, indiferent cum va juca al doilea jucător, primul jucător câștigă. Scrieți programul astfel încât primul jucător să mute cu ajutorul programului, iar calculatorul să mute aleator de la stânga sau de la dreapta. La ieșire se va scrie suma obținută de primul jucător, suma obținută de cea de al doilea și secvențele de mutări sub forma unor șiruri cu caracterele S pentru stânga și D pentru dreapta