

Probleme Curs 10 SDA Greedy

Probleme propuse pentru greedy. Unele din problemele propuse sunt preluate de pe *GeeksForGeeks*.

Probleme trasare

Problema 1 Se da un rucsac a carui capacitate este $W = 100$, si obiectele din tabelul de mai jos, fiecare avand greutatea w_i si profitul p_i . Fiecare obiect poate fi ales in rucsac, sau nu (decizie binara). Ce obiecte vor fi selectate in solutia obtinuta de o tehnica greedy ce utilizeaza maximizarea densitatii de profit? Este aceasta solutie optima? Care este valoarea solutiei optime?

Obiect	A	B	C	D	E	F
Greutate	75	60	25	20	10	5
Profit	25	12	12.5	4	10	2

Problema 2 Se da problema selectiei activitatilor, si doi algoritmi greedy pentru aceasta problema:

- (a) unul care foloseste ca euristica durata unei activitati – *MinDurationActivitySelector*(A, s, f)
- (b) celalalt care foloseste ca euristica timpul de finalizare – *EarlyFinishActivitySelector*(A, s, f)

Pentru activitatile $A = \{p, q, r, s, t, u, v, w, x, y, z\}$ cu urmasorii timpi de incepere si finalizare: $s = \{1, 3, 0, 5, 3, 5, 6, 8, 8, 2, 12\}$, respectiv $f = \{4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$, ce activitati vor fi selectate in urma aplicarii fiecaruia dintre acesti 2 algoritmi? E vreuna din solutii optima? Motivati.

Problema 3 Considerati mesaje care contin doar literele (C, E, T, O, R), cu urmatoarele probabilitati de aparitie: C: 0.27, E: 0.1, T: 0.18, O: 0.2 si R: 0.25.

- (a) Folosind algoritmul lui Huffman, construiti codificarea de lungime variabila, presupunand ca fiecare litera este codificata separat. Desenati arborele rezultat si dati codificarea pentru fiecare litera.
- (b) Decodificati mesajul: 1000010111010111

Author(s): Raluca Brehar, Camelia Lemnaru

- (c) Calculati lungimea asteptata a unui mesaj continand 100 de litere.

Probleme concepere algoritmi

Problema 4 Interclasare optimala:

Se dau dimensiunile a n siruri ordonate crescator. Sa se determine ordinea optima de interclasare a celor n siruri care sa asigure un numar total minim de operatii de transfer (copiere, atribuire) de elemente, stiind ca la un moment dat se pot interclasa numai doua siruri.

Problema 5 Colorarea grafurilor: Fie dat un graf neorientat cu n varfuri, sa se determine o colorare a varfurilor sale cu un numar minim de culori astfel incat oricare doua varfuri adiacente sa fie colorate diferit.

Problema 6 Fiind dat un sir de cifre cu valori de la 0 la 9 determinati suma minima a doua numere formate din cifrele din sirul dat. Toate cifrele sirului trebuie folosite pentru a forma cele doua numere.

1. Exemplu:

Se da sirul: $[6, 8, 4, 5, 2, 3]$; Suma 604 este formata din numerele 358 si 246 .

Se da sirul: $[5, 3, 0, 7, 4]$; Suma 82 este formata din numerele 35 si 047 .

Problema 7 Maximizarea valorii unei expresii: Se considera doua multimi de numere intregi nenule $A = a_1, a_2, \dots, a_m$ si $B = b_1, b_2, \dots, b_n$ cu $n \geq m$. Sa se determine o submultime $X = x_1, x_2, \dots, x_m$ a multimii B astfel incat expresia $E = a_1x_1 + a_2x_2 + \dots + a_mx_m$ sa fie maxima.

Example 1. Fie $A = \{100, 50, 20, 10, 5, 1, -10, -20, -30\}$ si $B = \{8, 5, 3, 1, -2, -10, -100\}$.
Expresia E este: $E = \{8 \times 100 + 5 \times 50 + 3 \times 20 + 1 \times 10 + (-100) \times (-30) + (-10) \times (-20) + (-2) \times (-10)\} = 4340$

Problema 8 Gara CFR cluj si numarul minim de platforme.

Stiind timpul de sosire si plecare a n trenuri din gara Cluj, determinati numarul minim de platforme necesare astfel incat sa nu fie niciun tren in asteptare pe o linie. Se dau ca intrare doua siruri reprezentand timpul de sosire si de plecare al trenurilor.

2. Exemplu:

sosire[] = 9:00, 9:40, 9:50, 11:00, 15:00, 18:00

plecare[] = 9:10, 12:00, 11:20, 11:30, 19:00, 20:00

Numar necesar de platforme: 3

Exista maxim 3 trenuri in acelasi timp in gara.

Problema 9 Problema calului pe o tabla de sah:

Se da o tabla de sah de dimensiune $n \times n$. Pe tabla este amplasat un cal la o pozitie (i, j) . Se cere sa se determina mutarile pe care le face calul astfel incat sa traverseze toata tabla de sah, fara a trece prin acelasi loc de doua ori.

3. Sugestie: 1. Daca pozitia calului este la celula (i, j) se determina toate mutarile posibile pe care le poate face din acea pozitie. La fiecare mutare posibila se calculeaza cate mutari mai poate face de acolo.

2. Se muta calul pe pozitia care furnizeaza un numar minim de pasi din mutarile posibile.

3. Continua conform algoritmului pana cand nu se mai gaseste un patrat unde sa se poata face o mutare.
