

Probleme Curs 9 SDA Backtracking

Probleme propuse pentru backtracking. Unele din problemele propuse sunt preluate de pe [GeeksForGeeks](#).

Probleme trasare

Problema 1 Problema plasarii celor n dame pe o tabla de sah se enunta in felul urmator: Se cere sa se plaseze n dame pe o tabla de sah de dimensiune $n \times n$ astfel incat nici o dama sa nu atace nici o alta dama. Pentru $n=5$, desenati spatiul de cautare care ar fi explorat de un algoritm bazat pe backtracking (presupuneti ca aranjarea se face cate o dama pe fiecare rand - i.e. row-wise, cautand coloanele corespunzatoare - valide) pentru gasirea primei solutii.

Problema 2 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). Cum arata spatiul de cautare explorat de o strategie de tip branch-and-bound pentru problema data?

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

Problema 3 Problema numararii restului: Avem o cantitate infinita de monede si bancnote de diferite valori - valorile acestora se cunosc. Se cere sa se returneze o anumita suma, A , folosind numarul minim de monede. Considerand o solutie bazata pe branch and bound la aceasta problema, aratati spatiul de cautare pentru valorile monedelor: $Values = \{1, 2, 5, 10\}$ si suma de returnat $A = 14$. Cum se exploreaza acest spatiu? Este necesara parcurgerea intregului spatiu in solutia bazata pe branch and bound? Argumentati.

Probleme concepere algoritmi

Problema 4 Aruncarea zarurilor:

Author(s): Raluca Brehar, Camelia Lemnar

- (a) Scrieti o functie `aruncaZaruri(int n)` care are ca parametru un numar intreg n , reprezentand nu numar de zaruri. Functia va afisa toate combinatiile posibile de valori care pot apare la aruncarea celor n zaruri.
 - (b) Scrieti o functie `zaruriSuma(int n, int sum)` care are un parametru in plus, sum si afiseaza doar combinatiile de zaruri care au suma egala cu sum .
-

Problema 5 Permutari: scrieti o functie `permuta(string a)` care are ca si parametru un sir de caractere si afiseaza toate permutarile posibile ale literelor din cuvantul dat. Permutarile pot fi in orice ordine.

Problema 6 Combinari: scrieti o functie `combinari(string s, int k)` care afiseaza toate cuvintele ce se pot forma din literele cuvantului s si au exact k litere.

Problema 7 Backtracking in plan: un soricel se gaseste intr-un labirint de forma dreptunghiulara cu m linii si n coloane. Peretii sunt marcati cu 1 si culoarele cu 0. Se cunosc coordonatele initiale ale soricelului: Li , Ci . Sa se determine toate posibilitatile pe care le are soricelul pentru a iesi din labirint. Soricelul poate avansa pe 4 directii cate o celula (sus, dreapta, jos, stanga). Traseul soricelului va fi retinut de un vector cu doua campuri: coordonatele x si y .

Problema 8 Romeo si Julieta se gasesc intr-un labirint (se cunosc culoarele si peretii si coordonatele celor doi indragostiti). (a) Exista posibilitatea ca Romeo sa ajunga la Julieta? (b) in cazul in care cei doi se indreapta simultan unul catre celalalt pentru fiecare solutie se va afisa locul intalnirii (coordoanatele celulelor alaturate sau celulei comune de intalnire)
