

Seminar 12

Backtracking și Branch-and-bound.

Ștefan Ciobâcă, Dorel Lucanu
Universitatea “Alexandru Ioan Cuza”, Iași

Săptămâna 16 Mai - 20 Mai 2016

1. Câte noduri apar în arborele “pruned” pentru formula $f = (x_1 \vee x_2) \wedge (\neg x_2 \vee x_3) \wedge \neg x_4$?
2. Scrieți în Alk un algoritm backtracking pentru problema 3-CNF-SAT, așa cum a fost formalizată în Seminarul 4 (problema 7).
3. Implementați în Alk un algoritm backtracking pentru problema celor n regine.
4. Proiectați un algoritm de tip backtracking pentru problema “submulțime de sumă dată”. Prin proiectarea se înțelege inclusiv specificarea următoarelor informații:
 - (a) cum este reprezentată o soluție
 - (b) care sunt soluțiile parțiale
 - (c) care sunt succesorii direcți ai unei soluții parțiale
 - (d) când o soluție parțială este viabilă
5. Proiectați un algoritm de tip backtracking pentru problema “submulțime de sumă dată”.
6. Proiectați un algoritm de tip backtracking pentru problema Sudoku.
7. Proiectați un algoritm de tip backtracking pentru problema Nonogram.
8. Proiectați un algoritm de tip backtracking pentru problema circuitului hamiltonian.
9. Implementați în Alk algoritmul branch-and-bound pentru problema discretă a rucsacului.
10. Proiectați un algoritm de tip branch-and-bound pentru problema 15-puzzle (în general, $(n^2 - 1)$ -puzzle). Specificați ce se înțelege prin soluție parțială, cum se calculează costul unei soluții parțiale, cum se poate mărgini inferior costul extensiei unei soluții parțiale și care sunt succesorii direcți ai unei soluții parțiale.
11. Proiectați un algoritm de tip branch-and-bound pentru problema Traveling Salesperson.
12. Proiectați un algoritm de tip branch-and-bound pentru problema Maximum Independent Set.