

Algoritmi in podatkovne strukture 1

Visokošolski strokovni študij Računalništvo in informatika

**Sestopanje in
razveji in omeji**



Sestopanje





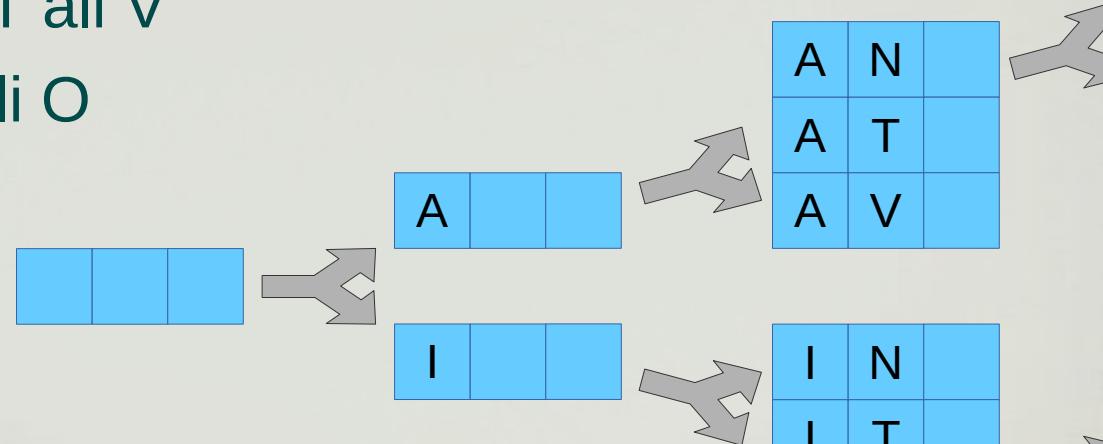
Sestopanje

- Ideja metode
 - rešitev generiramo po delih
 - za vsak del preizkusimo vse možnosti
 - izberemo eno možnost in stopimo v naslednji del
 - kot ni več možnih izbir, sestopimo v predhodni del
 - delna in celovita rešitev
 - pri izboru zadnjega dela dobimo celovito rešitev
- Uporaba
 - temeljito preiskovanje vseh možnih rešitev



Sestopanje

- Primer:
 - rešitev iz 3 delov
 - 1. del: A ali I
 - 2. del: N, T ali V
 - 3. del: A ali O



A	N	A
A	N	O
A	T	A
A	T	O
A	V	A
A	V	O

I	N	A
I	N	O
I	T	A
I	T	O
I	V	A
I	V	O

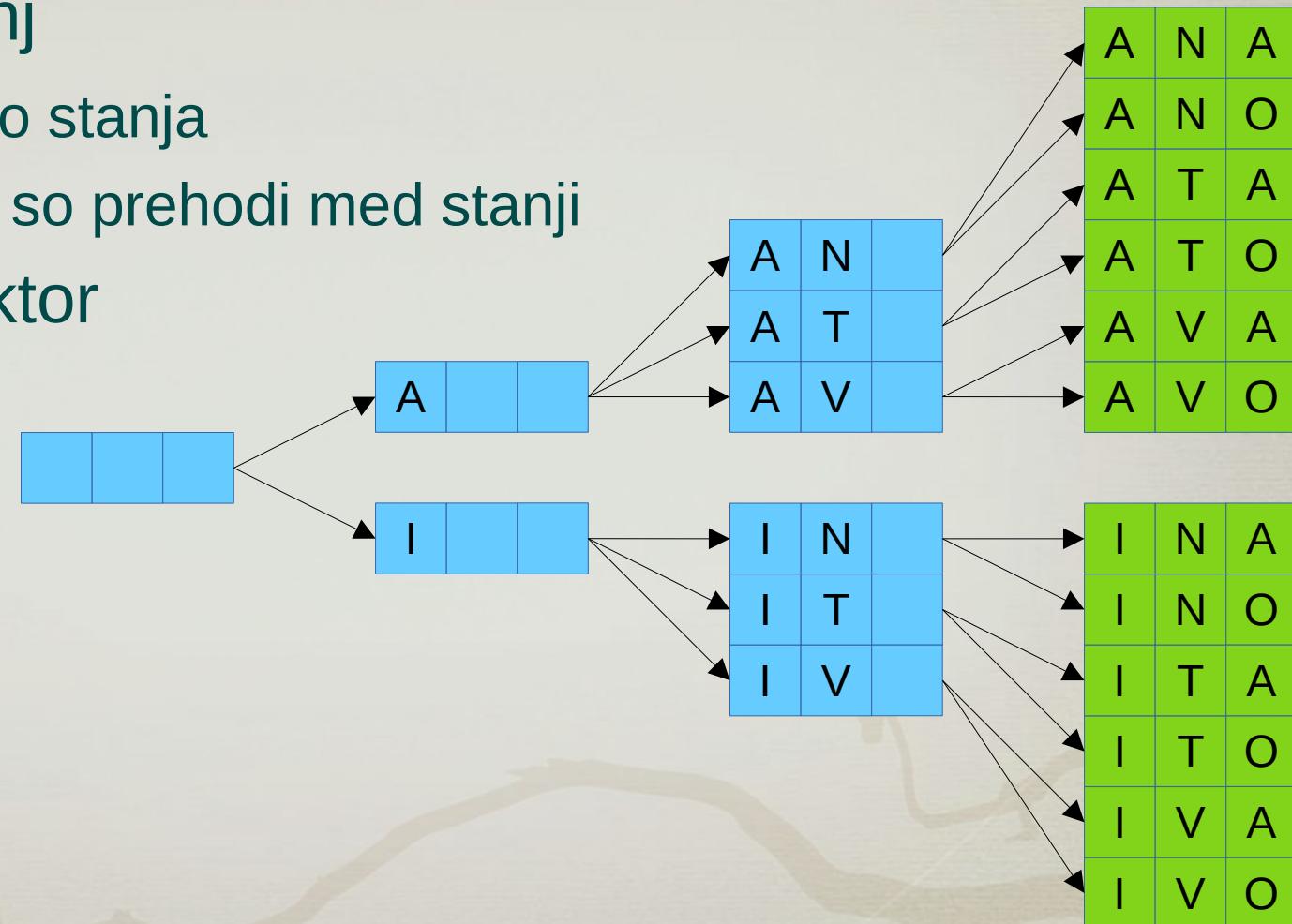
Sestopanje

- Drevo prostora stanj

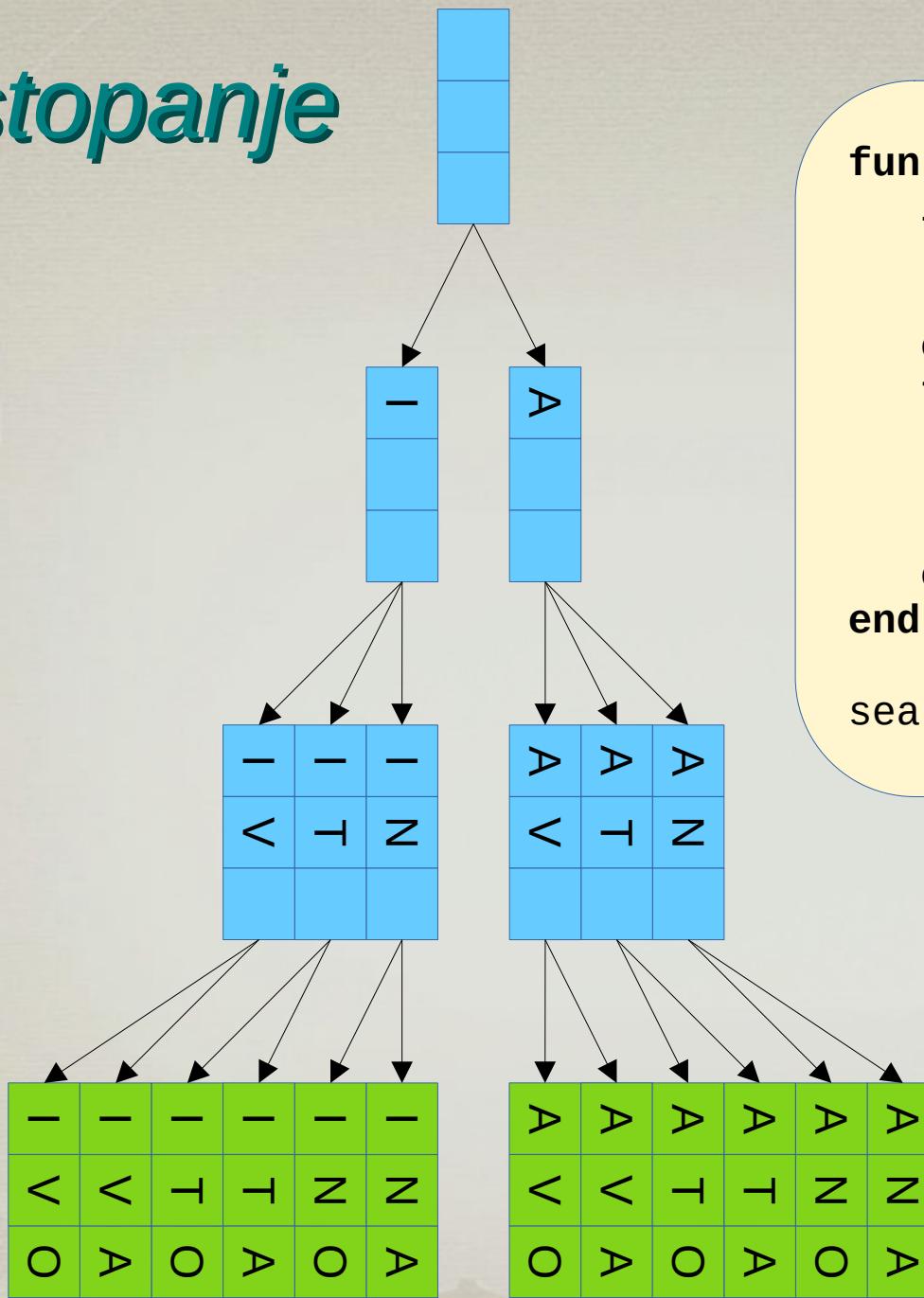
- prostor stanj

- vozlišča so stanja
 - povezave so prehodi med stanji

- vejitveni faktor



Sestopanje



```
fun search(S, k) is
    if complete(S, k) then
        print(S)
        return
    end if
    for c in candidates(S, k) do
        S.add(c)
        search(S, k + 1)
        S.remove(c)
    end for
end fun

search([], 0)
```

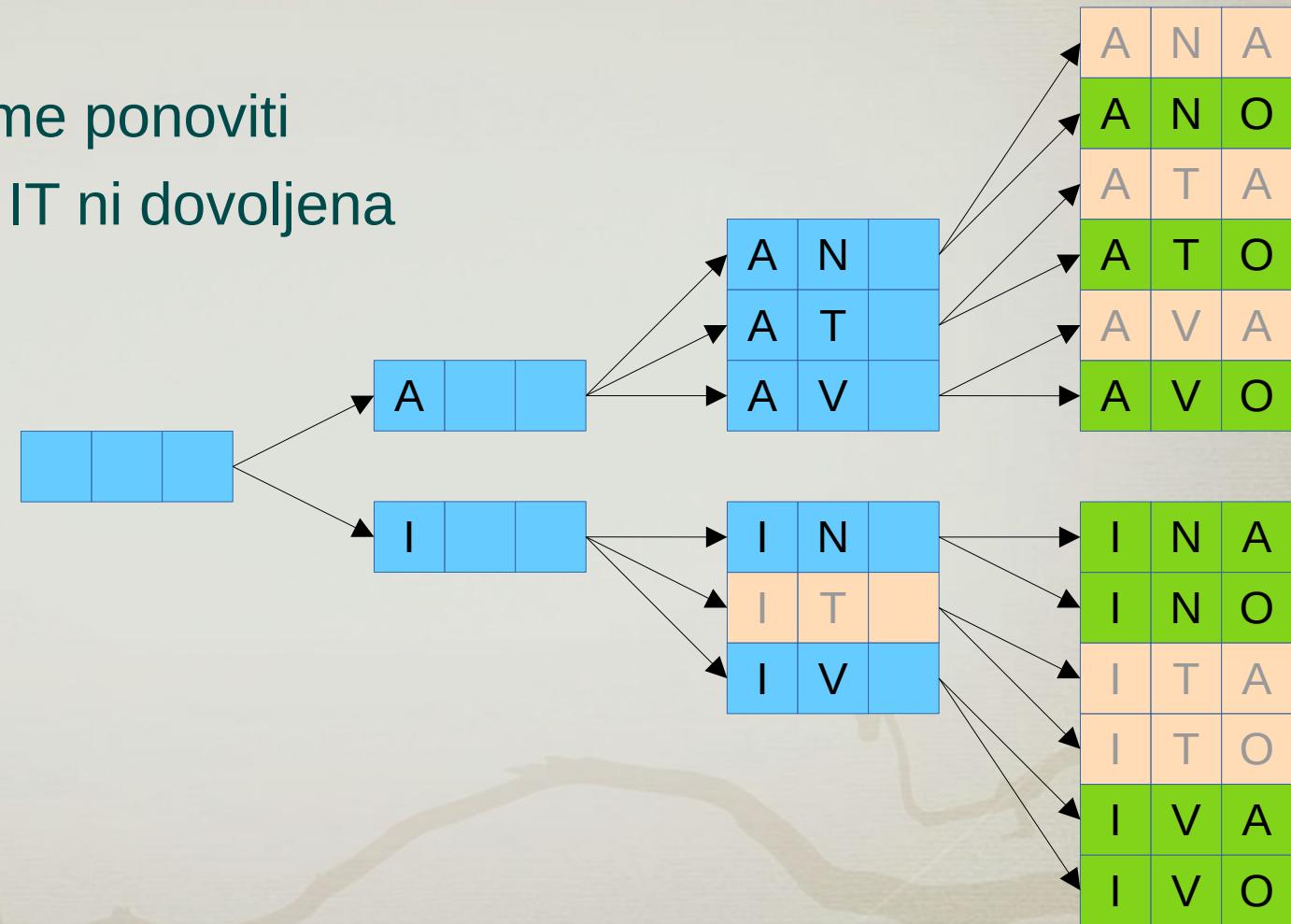
Sestopanje

- Rezanje (*pruning*) drevesa
 - brez rezanja
 - s *slepim* izbiranjem vseh možnosti se sestopanje običajno izrodi v izčrpno preiskovanje
 - rezanje
 - upoštevanje le **dopustnih** izbir
 - glede na omejitve naloge problema
 - glede na trenutno delno rešitev

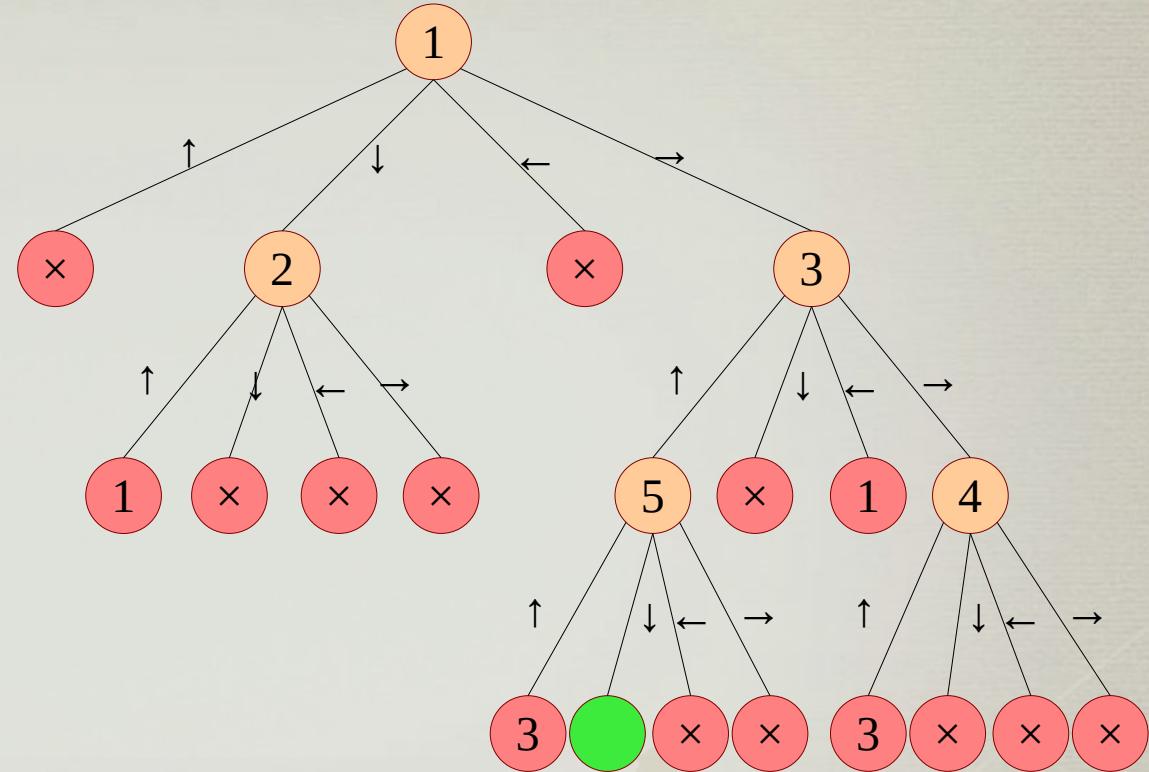
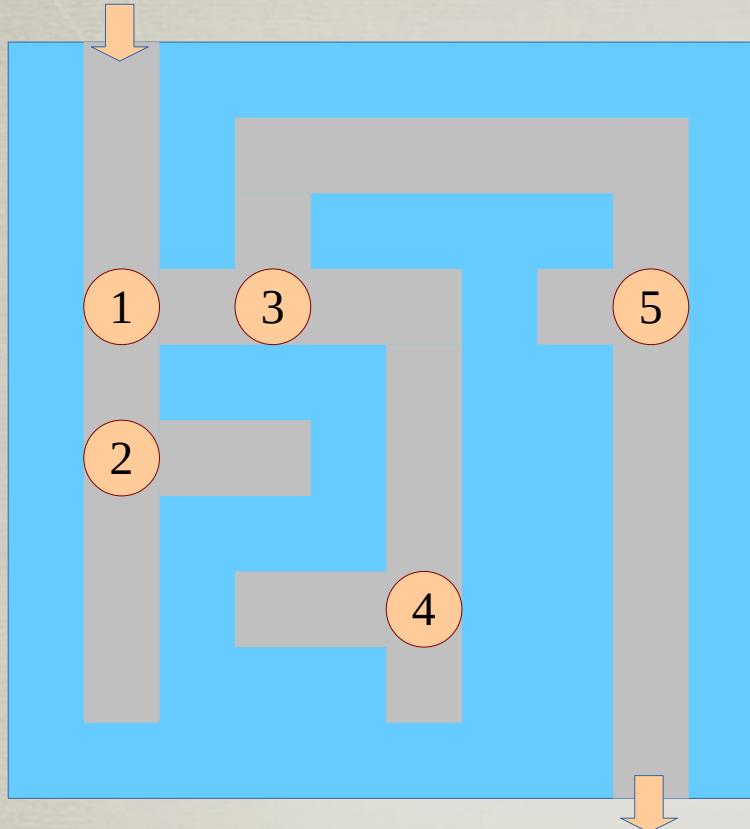


Sestopanje

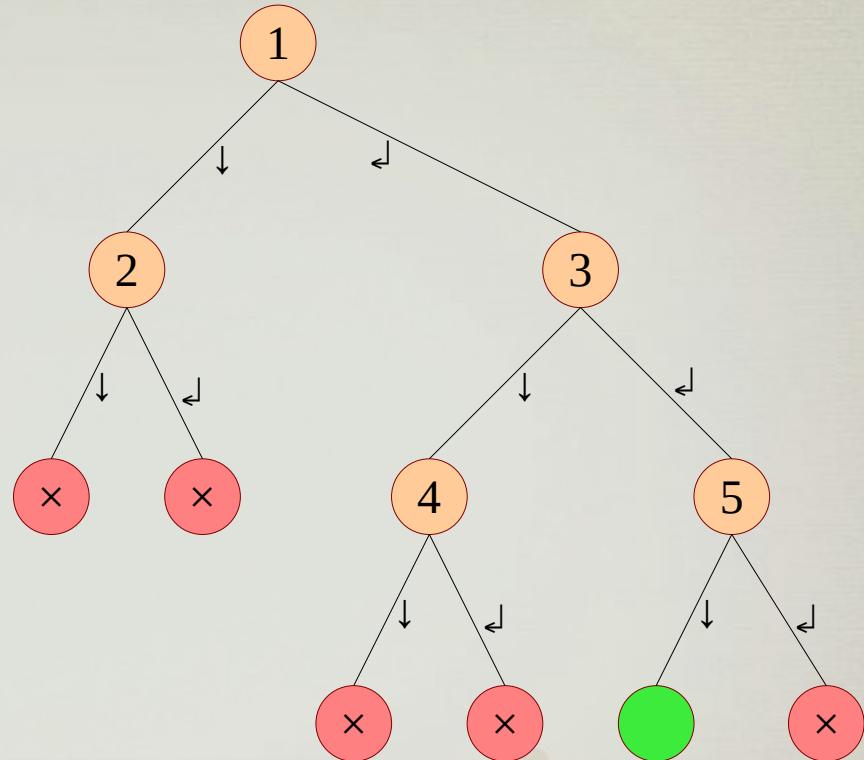
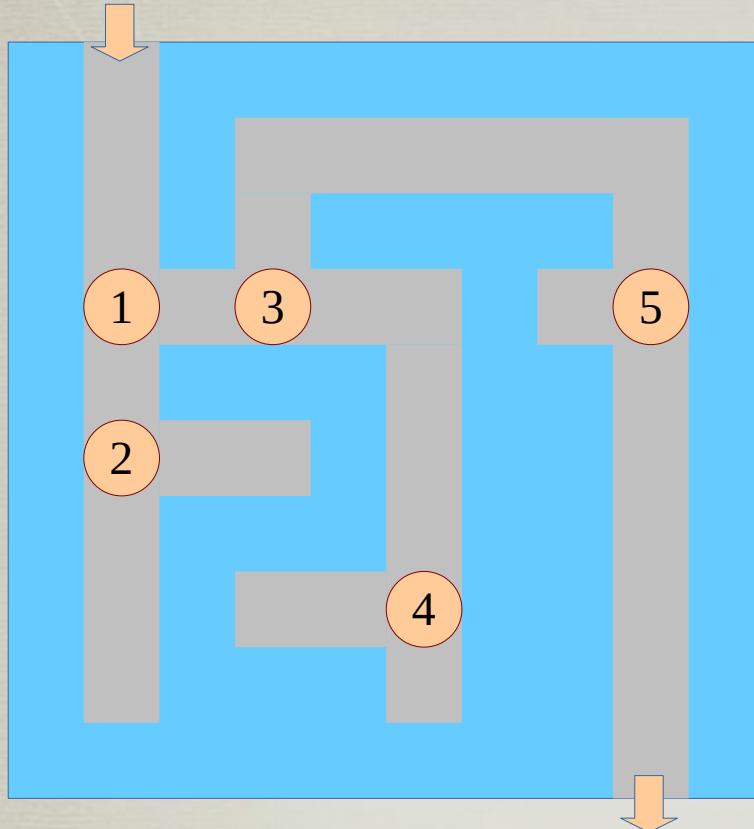
- Primer (nadaljevanje)
 - omejiteve
 - A se ne sme ponoviti
 - predpona IT ni dovoljena



Labirint

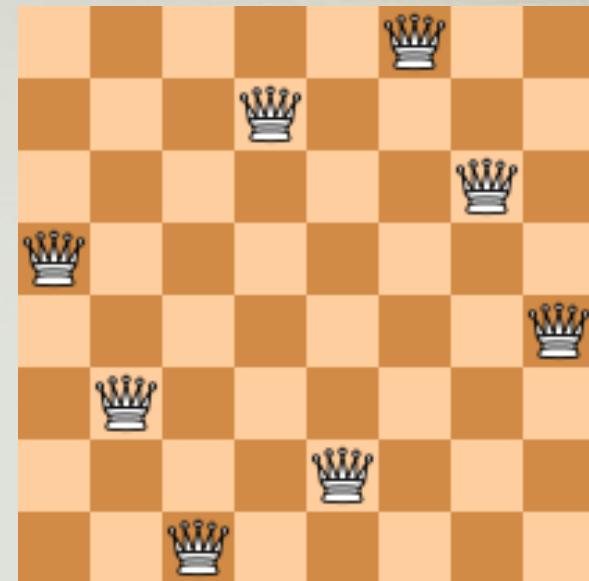


Labirint



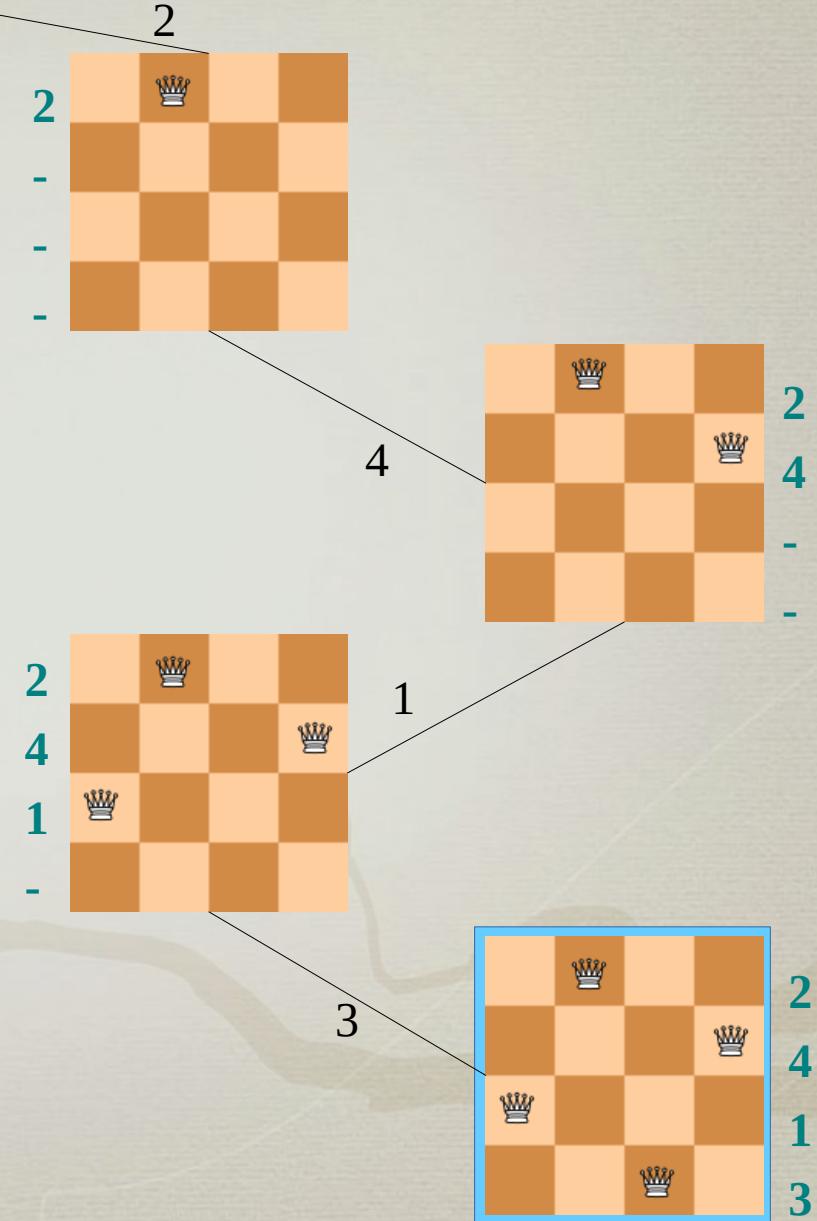
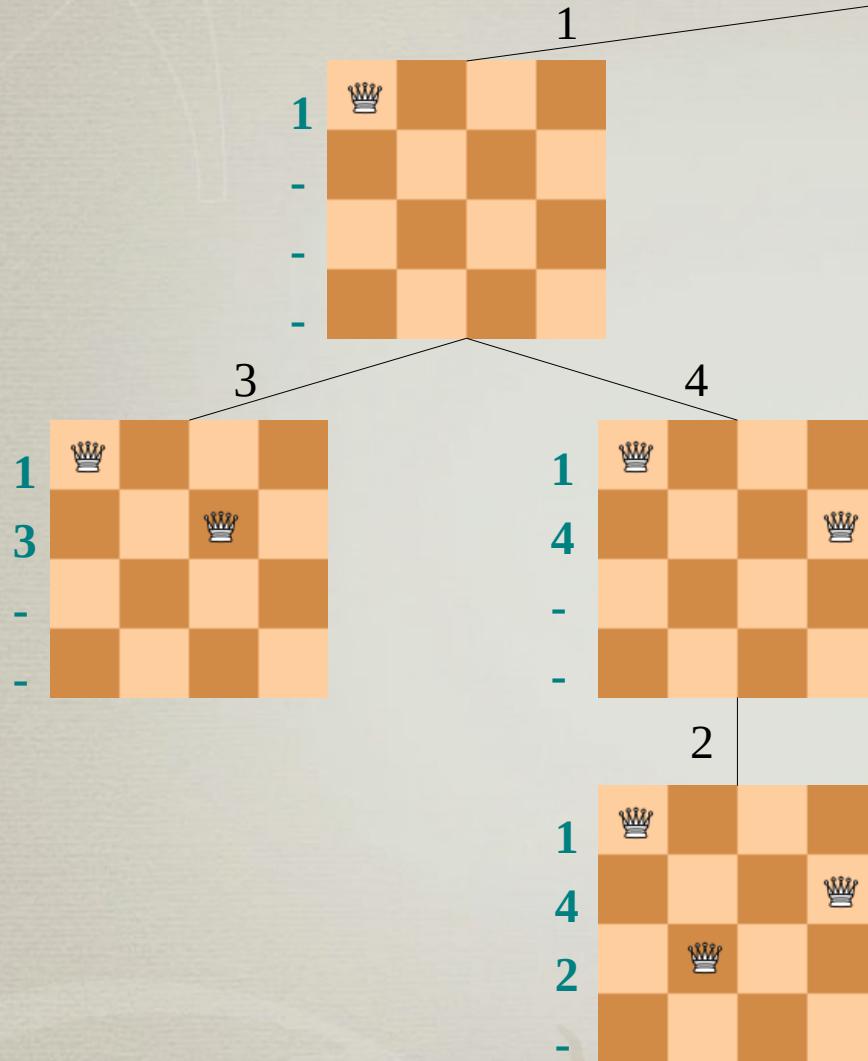
Kraljice na šahovnici

- Opis problema
 - šahovnica velikosti $n \times n$
 - postavi n kraljic tako, da se med seboj ne napadajo
 - število različnih rešitev



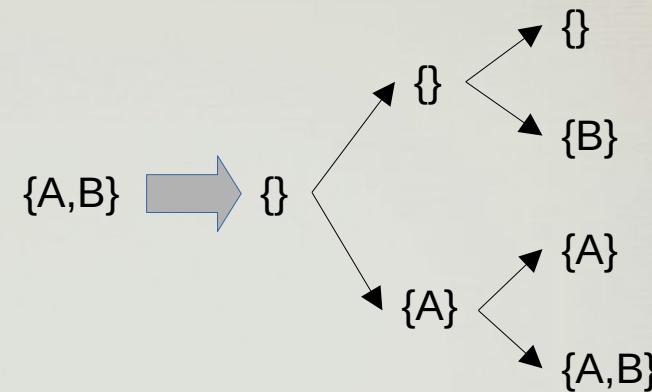
Kraljice na

šahovnici



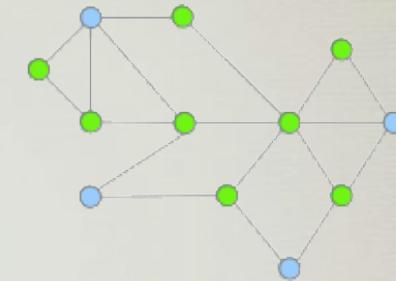
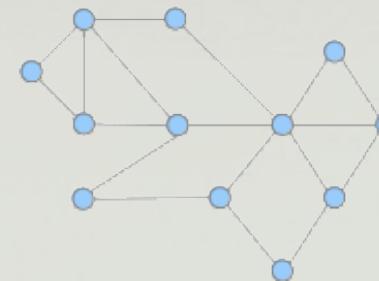
Kombinatorični objekti

- Iz danega seznama generiraj vse
 - podmnožice
 - kombinacije
 - permutacije



Vozliščno pokritje

- Vozliščno pokritje
 - izbrano je vsaj eno krajišče vsake povezave



- Najmanjše vozliščno pokritje
 - najmanjša podmnožica vozlišč, ki je vozliščno pokritje

Razvejji in omeji



Razveji in omeji

- Ideja metode
 - za optimizacijske metode
 - kriterijska funkcija vrne vrednost rešitve
 - iščemo optimalno rešitev (minimizacija, maksimizacija)
 - preiskovanje drevesa prostora stanj
 - v širino: obiskujemo najbolj obetavna vozlišča
 - v globino: podobno sestopanju z rezanjem
 - najboljše vozlišče (best-first search)
 - *razveji* se nanaša na obiskovanje vozlišč
 - *omeji* se nanaša na rezanje

Razvejji in omeji

- Ideja metode
 - meja oz. potencial kriterijske funkcije
 - mejo izračunamo v vsakem vozlišču drevesa
 - minimizacija: spodnja meja (LB, lower bound)
 - maksimizacija: zgornja meja (UB, upper bound)
 - dodatna evidenca
 - vrednost trenutno najboljše najdene rešitve (BEST)
 - rezanje
 - maksimizacija: če $UB \leq BEST$, potem režemo
 - minimizacija: če $LB \geq BEST$, potem režemo

0/1 nahrbtnik

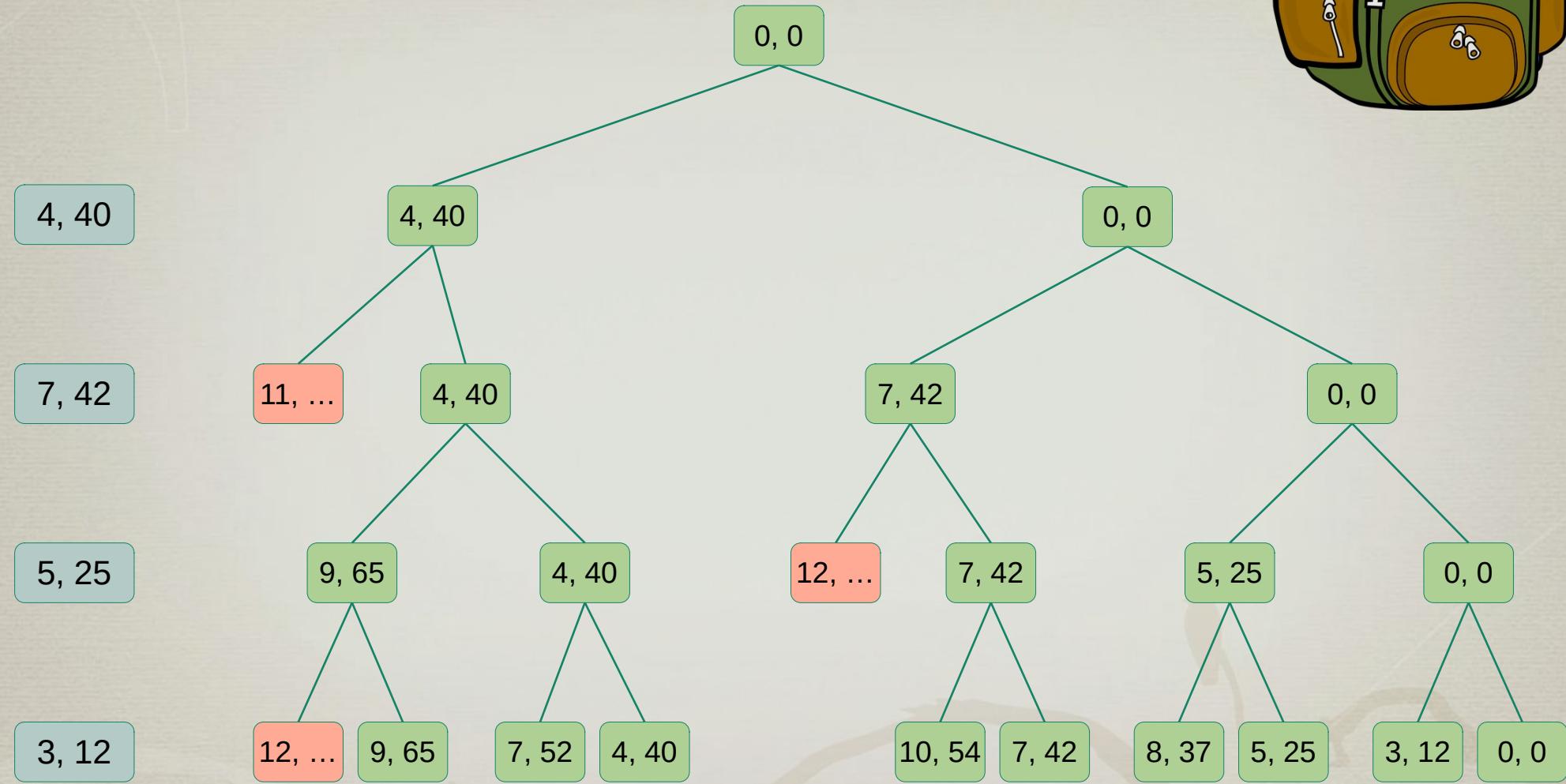
- Problem
 - danih je n predmetov
 - predmet i ima težo w_i in vrednost v_i
 - poišči nabor S predmetov, ki ne presegajo teže W in je njihova skupna vrednost čim večja



$$\sum_{x \in S} w_x \leq W$$

$$\max i \sum_{x \in S} v_x$$

0/1 nahrbtnik – sestopanje



0/1 nahrbtnik – razvejji & omeji

