Algoritmi in podatkovne strukture 1

Visokošolski strokovni študij Računalništvo in informatika

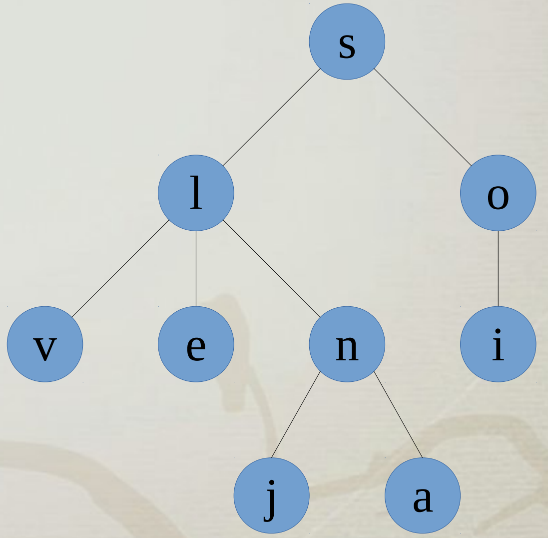
Ukoreninjena drevesa

Jurij Mihelič, UniLj, FRI

Drevesa

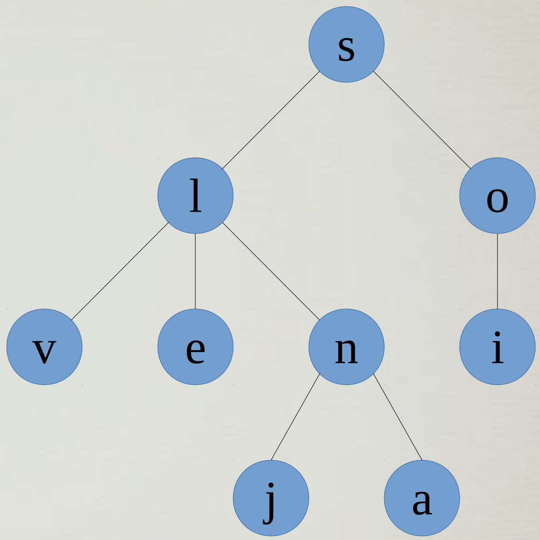
Drevesa

- Osnovni pojmi
  - drevo s korenom (ukoreninjeno drevo)
  - vozlišče, koren, povezava, element
  - notranje in končno (zunanje) vozlišče, list
  - starš, otrok, prednik, potomec
- Pot
  - povezave od izvora do cilja
  - dolžina poti = št. povezav



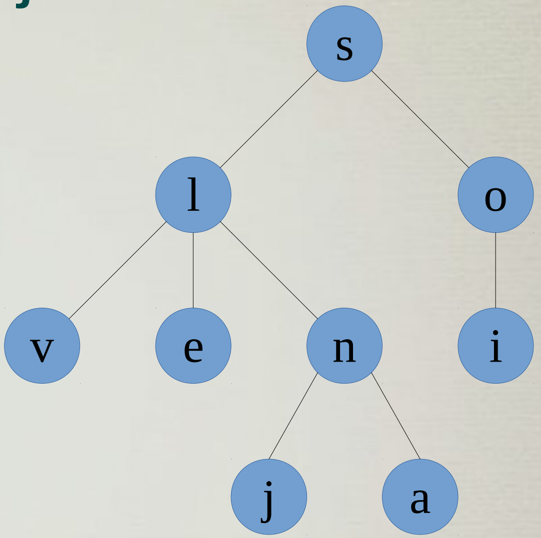
Drevesa

- Poddrevo
  - vozlišče in **vsi** njegovi potomci
  - vozlišče je koren poddrevesa
  - poddrevo je drevo
- Gozd
  - množica dreves
- Urejeno in neurejeno drevo
  - glede na vrsti red otrok
  - glede na urejenost starš / otrok



Drevesa

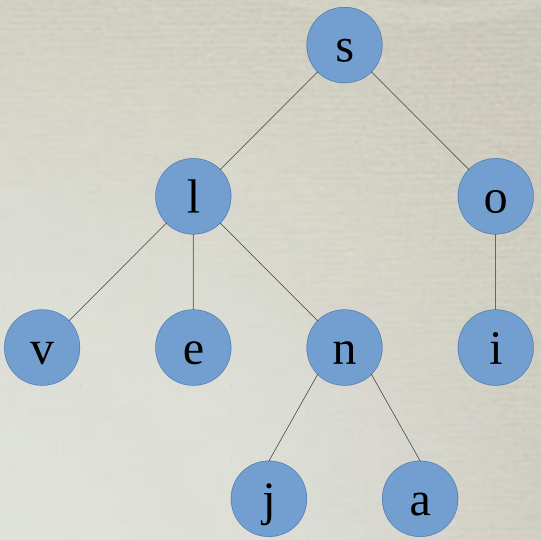
- Definicija
  - Ukoreninjeno drevo $T=(V,E,r)$ sestoji iz
    - končne množice vozlišč V in
    - končne množice povezav E
  - pri čemer
    - je eno vozlišče koren r *(root)*
    - ima vsako vozlišče 0 ali več otrok
    - potomci korena razpadejo na *disjunktno* unijo poddreves



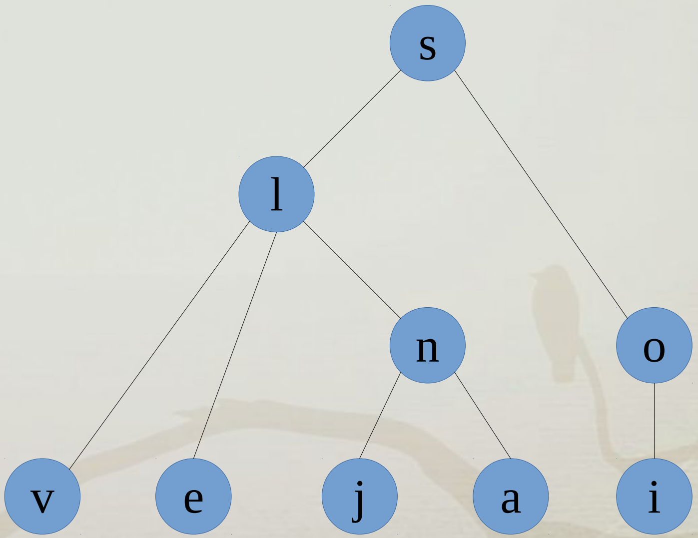
- Grafična ponazoritev
  - povezan acikličen graf
  - po nivojih

Lastnosti

- Globina vozlišča
  - dolžina poti od korena do vozlišča
  - nivo: vozlišča na isti globini

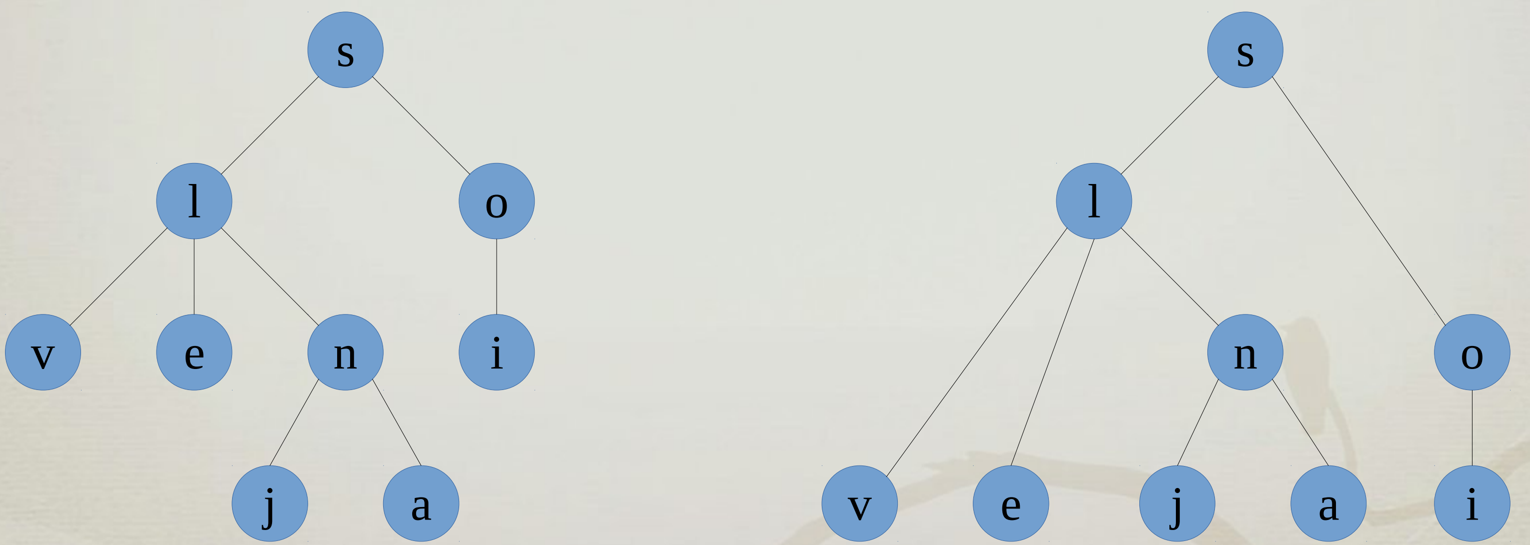


- Višina vozlišča
  - dolžina najdaljše poti od vozlišča do lista



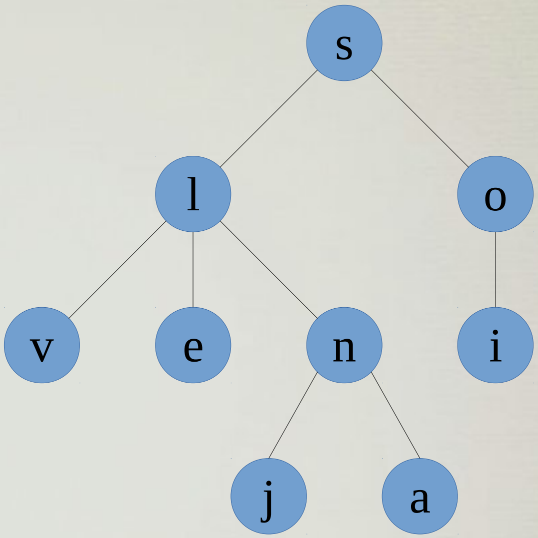
Lastnosti

- Višina (oz. globina) drevesa
  - je enaka višini korena
  - oz. dolžina najdaljše poti od korena do lista



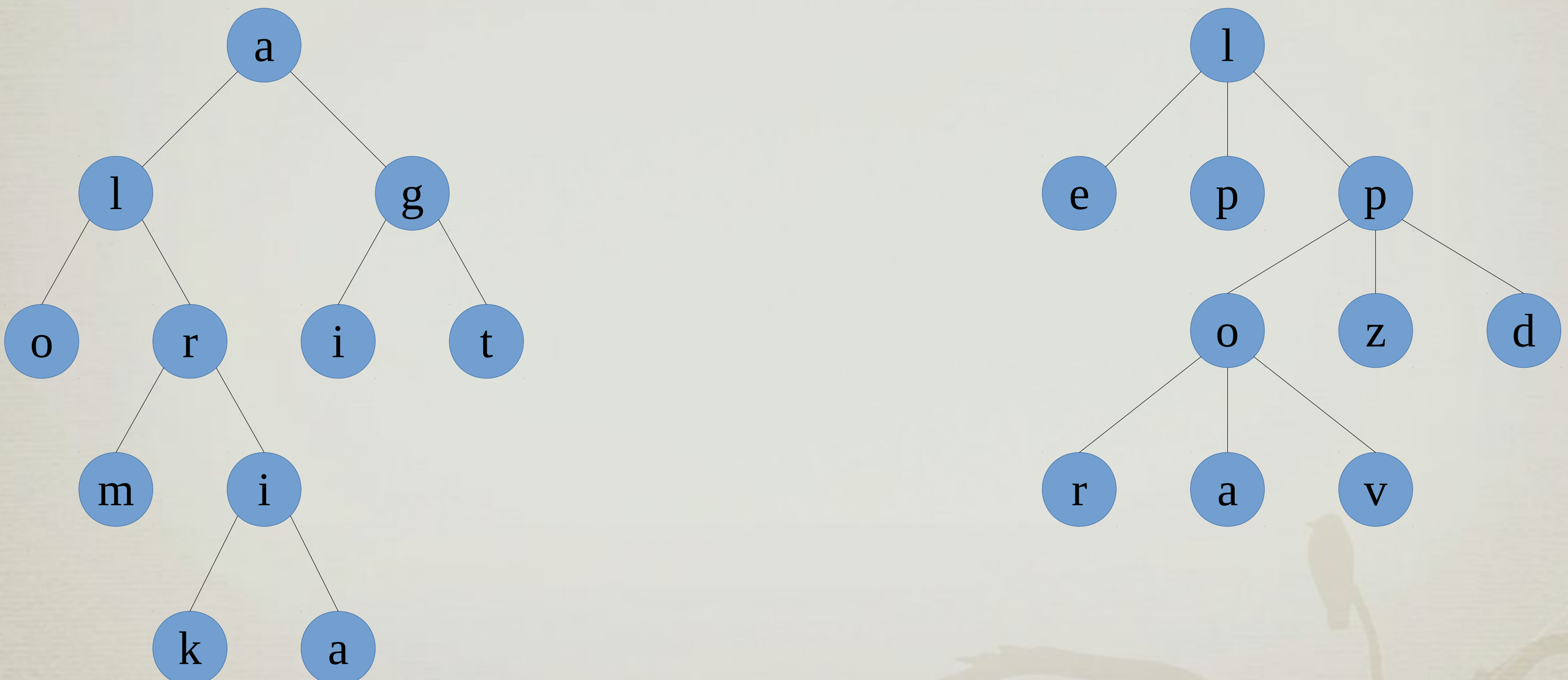
Lastnosti

- Stopnja (*degree*) vozlišča
  - št. otrok (oz. poddreves)
- Stopnja drevesa
  - največja stopnja vozlišča
  - dvojiško (*binary*) drevo
  - trojiško (*ternary*) drevo
  - d-tiško (d-*ary*) drevo
- Polno (*full*) vozlišče
  - stopnja vozlišča je enaka stopnji drevesa



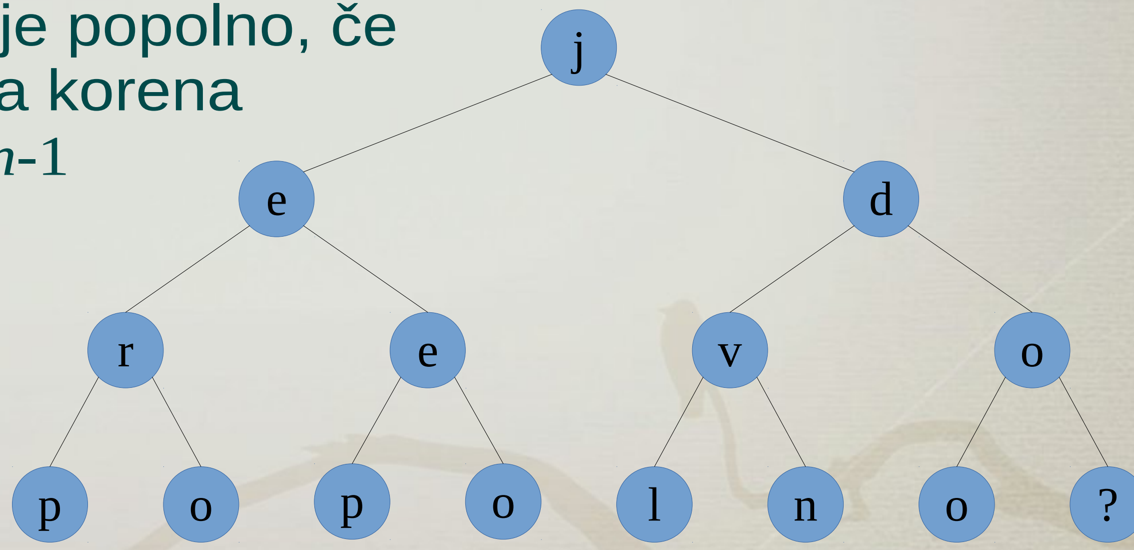
Lastnosti

- Polno drevo
  - vsa notranja vozlišča so polna



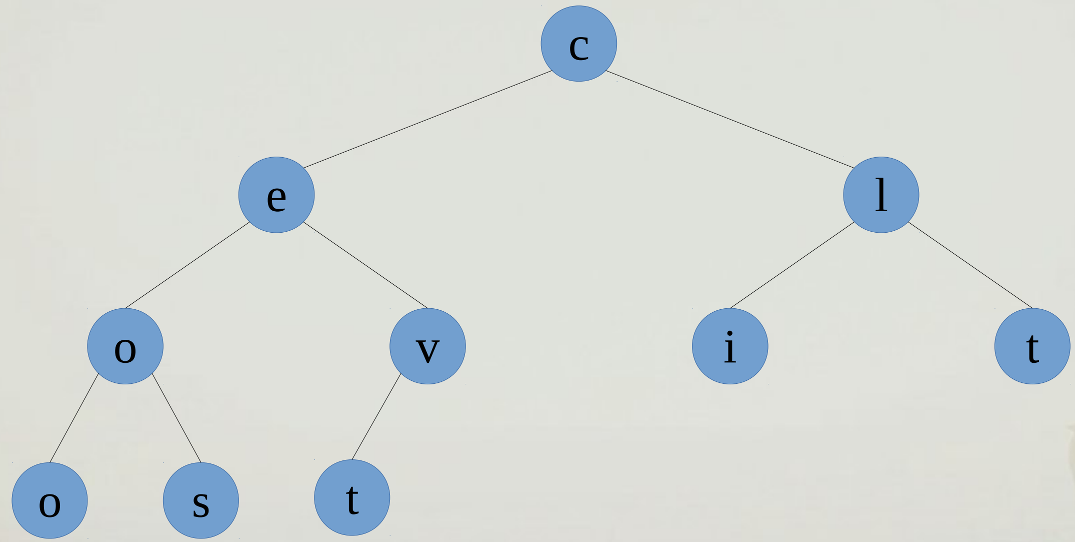
Lastnosti

- Popolno (*perfect*) drevo
  - polno drevo
  - vsi listi so na istem nivoju
  - rekurzivna definicija
    - drevo višine 0 je popolno
    - drevo višine $h > 0$ je popolno, če so vsa poddrevesa korena popolna in višine h-1



Lastnosti

- Celovito (*complete*) drevo
  - vsi nivoji (razen morda zadnji) so polni
  - vsi listi na zadnjem nivoju so levo



Lastnosti

- Algoritmi
  - štetje vozlišč
  - štetje listov
  - štetje notranjih vozlišč
  - globina vozlišča
  - višina vozlišča
  - stopnja drevesa
  - vsota stopenj vozlišč

Drevesni obhodi

- Sistematičen obisk vseh vozlišč drevesa
- Vrste obhodov (*traversal*)
  - premi obhod (*preorder*)
  - obratni obhod (*postorder*)
  - vmesni obhod (*inorder*)
    - le za dvojiška drevesa
  - obhod po nivojih (*level order*)

Drevesni obhodi

- Premi (direktni, neposredni) obhod drevesa
  - koren nato otroci
- Ideja algoritma
  - obdelaj koren drevesa
  - obhodi poddrevesa otrok korena

Drevesni obhodi

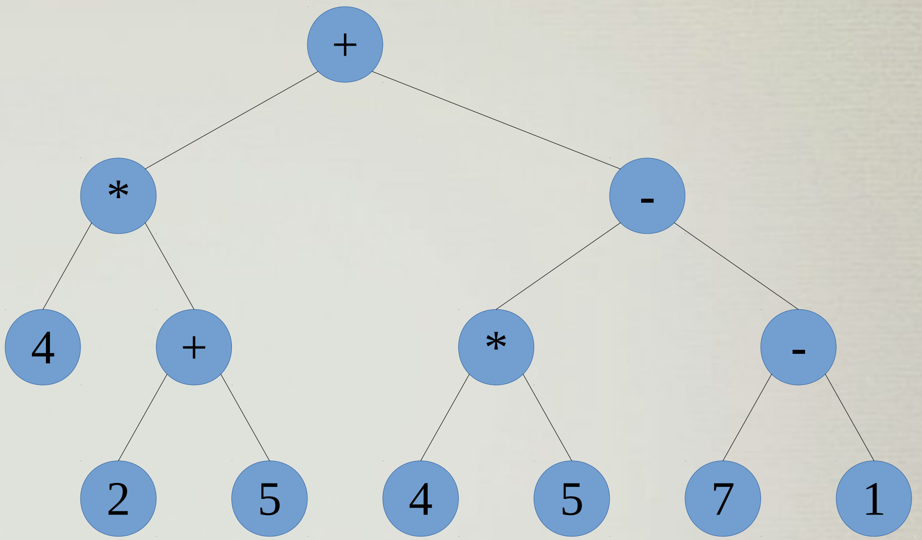
- Obratni obhod
  - otroci nato koren
- Ideja algoritma
  - obhodi poddrevesa otrok korena
  - obdelaj koren drevesa

Drevesni obhodi

- Vmesni obhod
  - le na dvojiških drevesih
  - levo poddrevo, koren, desno poddrevo
- Ideja algoritma
  - obhodi poddrevo levega otroka
  - obdelaj koren
  - obhodi poddrevo desnega otroka

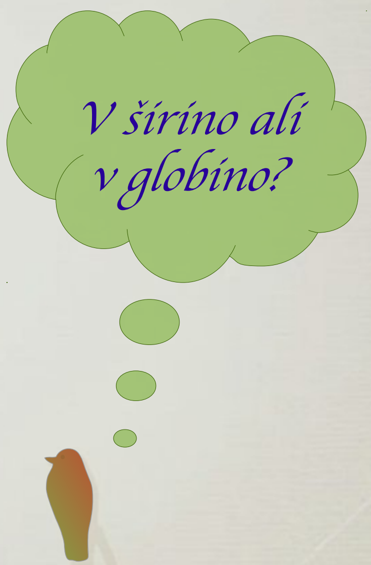
Drevesni obhodi

- Drevo za aritmetični izraz
  - 4 * (2+5) + (4*5 - (7-1))
- Premi obhod
  - +*4+25-*45-71
- Obratni obhod
  - 425+*45*71--+
- Vmesni obhod
  - ((4*(2+5))+((4*5)-(7-1)))



Drevesni obhodi

- Obhod po nivojih
  - zaporedoma obdelujemo nivoje
- Ideja algoritma
  - otroke shranjujemo v zbirko
  - katero zbirko uporabiti?



Predstavitev dreves

- S kazalci
  - otroci
    - zaporedje kazalcev na otroke
    - binarno drevo: kazalca na levega in desnega otroka
  - prvi otrok in sorojenci (*sibling*)
    - kazalec na prvega otroka
    - vsak otrok ima kazalec na naslednjega sorojenca
  - starš
    - kazalec na starša
    - za *neurejena* drevesa
    - za *urejena* drevesa s navadno kombinira s prejšnjima dvema metodama

Predstavitev dreves

- V polju (implicitna predstavitev)
  - dvojiška drevesa
    - vozlišče z indeksom i
    - otroka: $l = 2i+1$, $r = 2i+2$
    - starš: $p = \lfloor (i-1) / 2 \rfloor$
  - d-tiška drevesa
    - otroci: indeks j-tega otroka $= d \cdot i + 1 + j$
    - starš: $p = \lfloor (i-1) / \mathrm{d} \rfloor$
    - otroci so torej na indeksih od $d \cdot i + 1$ do $\mathrm{d} \cdot i + d$
  - učinkovitost predstavitve
    - kapaciteta polja in velikost drevesa
    - izrojena in celovita drevesa

Predstavitev dreves

- Celovita (dvojiška) drevesa v polju
  - `items` ... polje elementov
  - `last` ... indeks zadnjega
  - notranja vozlišča: prvih $\lfloor n/2 \rfloor$ elementov
  - listi: zadnjih $\lceil n/2 \rceil$ elementov