

# Algoritmi in podatkovne strukture – 2

## Prvi kolokvij (2014/15)

Kolokvij morate pisati posamič. Pri reševanju je literatura dovoljena. Pri odgovarjanju bodi natančni in: (i) odgovarjajte *na zastavljena* vprašanja; in (ii) odgovorite na *vsa* zastavljena vprašanja – če boste odgovarjali na vsa vprašanja, lahko dobite dodatne točke.

Čas pisanja izpita je 60 minut.

Veliko uspeha!

| NALOGA | TOČK | OD TOČK | NALOGA | TOČK | OD TOČK |
|--------|------|---------|--------|------|---------|
| 1      |      |         | 3      |      |         |
| 2      |      |         | 4      |      |         |

IME IN PRIIMEK: \_\_\_\_\_

ŠTUDENTSKA ŠTEVILKA: \_\_\_\_\_

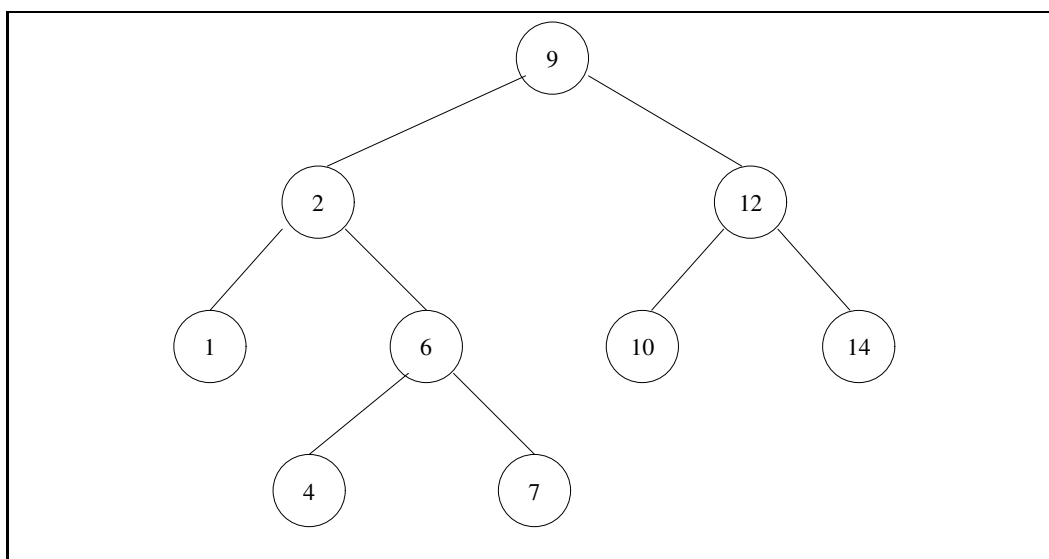
DATUM: \_\_\_\_\_

PODPIS: \_\_\_\_\_

**1. naloga:**

## VPRAŠANJA:

1. Koliko elementov najmanj je v rdeče-črnem drevesu višine<sup>1</sup> 5? Utemeljite odgovor.
2. Ali je lahko drevo na sliki sl. 1 rdeče-črno drevo? Utemeljite odgovor.

**Slika 1:** Primer drevesa.

3. V 2-3-4 (TTF) drevo po vrsti vstavljajte elemente 1, 2, 3, ... i.) Kako se drevo spreminja? Kaj opazite? ii.) Recimo, da smo v drevo vstavili po vrsti elemente od 1 do n. Kako izgleda drevo? Kako visoko je (čim natančneje in ne v  $O$  zapisu)?

**2. naloga:** Tokrat so klicali Petra z Univerze v Butalah. Na univerzi so sprejeli nova pravila, kdo je lahko član komisije za zagovor zaključnega dela. Za lažje razumevanje pravil, najprej definirajmo odnosa „soavtor“ in „posredni soavtor“. Recimo, da sta Janez in Peter avtorja članka. Potem je Janez Petrov soavtor in obratno. Odnos posredni soavtor je definiran tako, da je vsak soavtor avtorja tudi njegov posredni soavtor in vsak soavtor posrednega soavtorja je prav tako posredni soavtor prvega avtorja. Univerza sedaj zahteva, da član komisije ne sme biti posredni soavtor ne kandidata in ne njegovega mentorja.

Peter mora narediti aplikacijo, katera najprej prebere vse članke, ki naj se upoštevajo, ter z njih razbere avtorje. Nato uporabnika povpraša po kandidatu ter

<sup>1</sup>Drevo s samo enim elementom ima višino 0.

njegovem mentorju. Od tu sledijo poizvebe z imeni morebitnih članov komisije ter odgovori, ali ustrezajo zahtevam ali ne.

VPRAŠANJA:

1. Predlagajte in opišite učinkovito podatkovno strukturo, ki naj jo Peter uporabi v svoji aplikaciji.
2. Na univerzi so ugotovili, da baza člankov ni dokončna in lahko članke dodajamo. Ali lahko še vedno uporabljate enako podatkovno strukturo kot v prvem vprašanju? Predlagajte učinkovito rešitev ter utemeljite odgovor.
3. Kakšna je časovna in prostorska zahtevnost obeh vaših rešitev? Utemeljite odgovor. Upoštevajte tri vrste operacij: predprocesiranje, poizvedba in v drugem primeru tudi dodajanje.

**3. naloga:** *Kako težki so?* Otroci so se igrali na dvorišču naslednjo igro ugibanja. Ugibali so skupno težo določene podmnožice otrok in sicer je ta podmnožica definirana na podlagi višine otrok. Tako je bila poizvedba lahko: „Kolikšna je skupna teža vseh otrok, ki so višji od 112 cm in nižji od 127 cm?“ Recimo, da to poizvedbo poimenujemo `Teza(112, 127)`.

Recimo, da imamo naslednjo množico otrok s težami (prva številka predstavlja višino in druga teža): (120, 112), (128, 78), (153, 108), (98, 130), (117, 116), (98, 102), (149, 122), (129, 113), (116, 102) in (114, 103).

VPRAŠANJA:

1. Pri opisani množici otrok odgovorite na naslednje poizvedbe:  
`Teza(108, 141)`, `Teza(121, 151)`, `Teza(97, 114)`,  
`Teza(149, 124)` in `Teza(124, 123)`.
2. Recimo, da je v množici  $n$  otrok. Opišite podatkovno strukturo, ki časovno kar se da učinkovito odgovarja na poizvebo `Teza()`. Utemeljite njeno pravilnost ter njeno prostorsko in časovno učinkovitost.
3. Množica otrok na dvorišču seveda ni stalna, ker jih mame kličejo na kosilo ali večerjo ali pa se pridružujejo skupini. Dodelajte podatkovno strukturo, da bo podpirala še operaciji

`Odsel(visina, teza)` in `Prisel(visina, teza)`.

Kakšna je časovna zahtevnost vseh treh vaših operacij? Kakšna je prostorska zahtevnost podatkovne strukture? Utemeljite odgovor.

**4. naloga:** *Kateri so najtežji?* Ponovno smo pri igri iz prejšnje naloge, le da tokrat težo pri vsakem otroku zmanjšamo za 111 enot in imamo sedaj pare: (120, 1), (128, -33), (153, -3), (98, 19), ..., (114, -8).

VPRAŠANJA:

1. Vsota tako spremenjenih tež med katerima višinama je največja? Opišite postopek, kako ste to izračunali.
2. Recimo, da imamo  $n$  otrok in da so vse njihove višine različne. Kakšna je časovna in prostorska zahtevnost vašega postopka iz prvega vprašanja? Ocenite zahtevnosti za posamezne korake (faze), če se seveda vaš postopek sestoji iz njih.
3. Če primerjamo teži dveh otrok, njuno razliko zapišemo
  - kot A, če je prvi več kot 20 enot lažji od drugega;
  - kot B, če je teža prvega 20 ali manj enot manjša od teže drugega;
  - kot C, če je teža drugega enaka teži prvega ali 20 ali manj enot manjša od teže drugega; in
  - kot D, če je teža drugega otroka več kot 20 enot manjša od teže prvega otroka.

Če se vrnemo k našemu primeru iz prejšnje naloge, lahko zapišemo množico kot DAACCACCB, če primerjamo paroma zaporedne otroke iz množice. Takšnemu zapisu rečemo karakteristika množice. Recimo, da imamo podani karakteristiki dveh množic  $C_1$  in  $C_2$ , potem njuno *podobnost* določa zaporedje črk, ki se pojavlja v obeh karakteristikah v enakem vrstnem redu, vendar so lahko vmes še druge črke. Naj bo  $C_1$  DAACCACCB in  $C_2$  DDACCBBCABCBCB, potem je primer podobnosti zaporedje črk ACACB (črke podobnosti so označene v obeh karakteristikah). Razdaljo med karakteristikama definiramo kot dolžino najdaljše podobnosti.

- i) Kakšno tehniko programiranja bi uporabili, da bi našli najdaljšo podobnost?
- ii) Skicirajte algoritem.