Algoritmi in podatkovne strukture – 2
Drugi kolokvij (2012/13)

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.

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: Peter Zmeda je slišal, da obstajajo različne vrste kopic kot izvedbe vrst s prednostjo. Tako je slišal, da obstajata Fibonaccijeva in binomska kopica.

VPRAŠANJA:

1. Nad Fibonaccijevo kopico po vrsti naredite naslednje operacije in sproti izrisujte podatkovno struktur (`I` pomeni vstavi, `M` minimum in `DM` zbriši najmanjši element):

   `I` 17, `I` 3, `I` 5, `I` 1, `M`, `I` 10, `DM`, `I` 4, `DM`, `DM`

2. Iste operacije izvedite še nad binomsko kopice ter ponovno sproti izrisujte izgled strukture.
3. Ali obstaja primer, ko je Fibonaccijeva kopica primernejša od binomske kopice in ali obstaja primer, kjer je binomska kopica primernejša od Fibonacijeve. Oba primera utemeljite.

2. naloga: Če imamo v računalniku poljubno besedilo, za shranjevanje posameznih črk besedila porabimo od 8-16 bitov – v vsakem primeru za vsako črko enako število bitov. Vendar se črke ne pojavaljajo enako pogosto. Kaj bi se zgodilo, če bi za kodiranje različnih črk uporabili različno število bitov?

Poglejmo si primer:

> Imamo besedilo: `aaabacbba`
>
> Imamo 3 črke. Za vsako črko potrebujemo po 2 bita (recimo: `a`=`00`, `b`=`01`, `c`=`10`), kar pomeni da za zakodiranje tega besedila potrebujemo $9 \cdot 2 = 18$ bitov: `000000010010010100`.
>
> Če pa `a` zapišemo kot bitni kodni niz `1`, `b` kot bitni kodni niz `01`, in `c` kot bitni kodni niz `00`, potem je koda besedila `1110110001011` in potrebujemo $5 \cdot 1 + 3 \cdot 2 + 1 \cdot 2 = 13$ bitov!

VPRAŠANJA:

1. Recimo, da je koda črke `x` kodni bitni niz w_x in koda črke `y` kodni bitni niz w_y. Kakšno razmerje *mora* veljati med kodnima nizoma w_x in w_y, da bo kodo $w_x w_y$ možno dekodirati kot `xy`?
2. Imamo abecedo $\Sigma = \{a_1, a_2, ..., a_n\}$ in besedilo $t[1..m]$, ki je sestavljeno iz črk naše abecede ($n << m$). (i) Zapišite algoritem, ki izračuna pogostnosti ν_i pojavljanja posameznih črk a_i v besedilu t. (ii) Kakšna je časovna in prostorska zahtevnost vašega algoritma?

3. Za abecedo Σ imamo izmerjene pogostnosti pojavljanja ν_i posameznih črk a_i v besedilu. (i) Opišite, kaj mora veljati za dolžine kodnih nizov w_i posameznih črk a_i; *ali* za več točk (ii) zapišite algoritem, kako izračunati kodne bitne nize za posamezne znake.

3. naloga: Tokrat so Petra poklicali organizatorji njujorškega maratona, ki imajo posebno željo. Namreč želijo postaviti spletno stran, preko katere bi lahko pregledovali trenutni vrstni red tekačev. Od Petra želijo, da njegova rešitev podpira čim učinkoviteje naslednje funkcije:

- `RaceStart()` – tekma se je pričela;
- `RaceEnded()` – tekma se je zaključila;
- `CurrentTime(who, time)` – ki osebi `who` popravi trenutni čas[1]; in
- `Place(who)` – ki vrne trenutno mesto tekmovalca `who`.

Seveda, spletna stran mora delovati tudi še po zaključku tekme.

VPRAŠANJA:

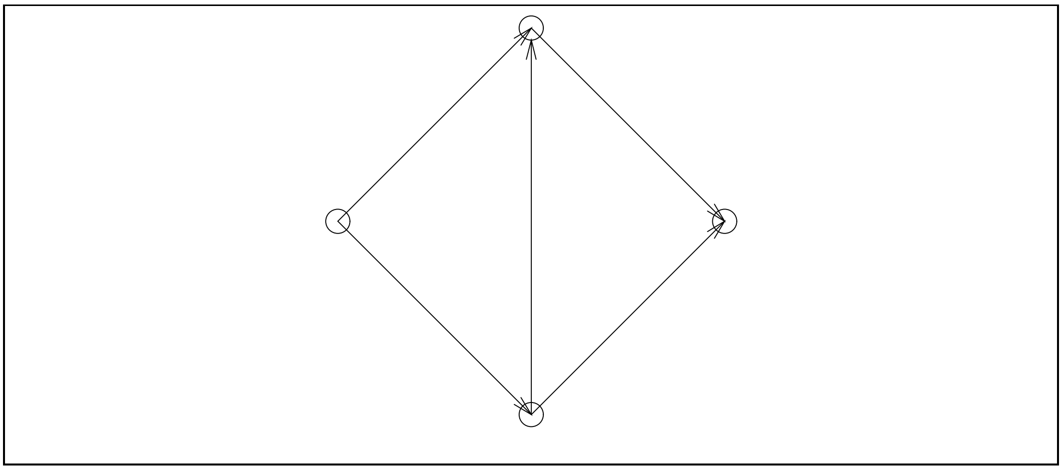
1. Opišite učinkovito podatkovno strukturo ali strukture, ki implementirajo zgornje operacije.

2. Nepovezano na zgornje vprašanje. Na predavanjih smo obravnavali Ford-Fulkersonov algoritem za izračunan največjega pretoka. Omenili smo tudi, da je časovna zahtevnost algoritma $O(mf)$, kjer je m število povezav in f velikost največjega pretoka, če so kapacitete vseh povezav cela števila. Narišite primer grafa, kjer bo algoritem dosegel to zgornjo mejo in utemeljite svoj odgovor.

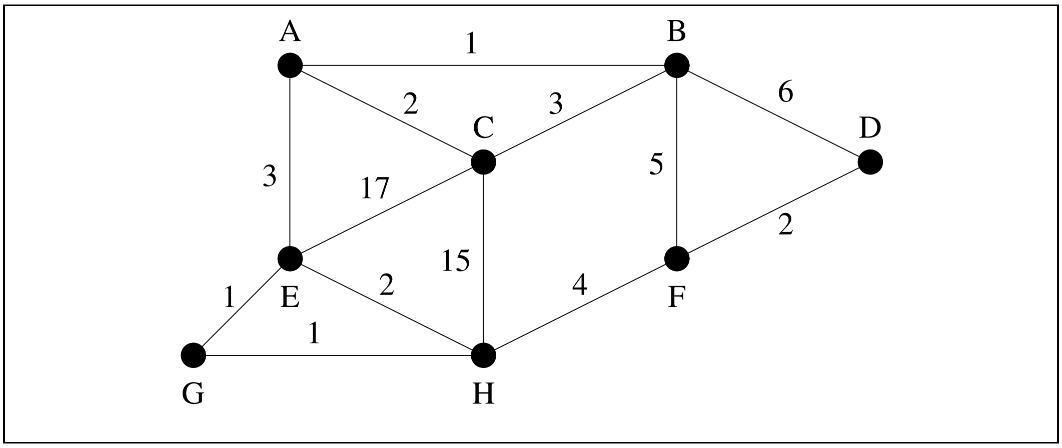
   NAMIG: Topologija grafa je na sl. 1 in samo dodajte kapacitete.

3. Imamo graf s sl. 2, v katerem poiščite Eulerjev sprehod. Eden od najbolj poznanih sprehodov v močno povezanem usmerjenem grafu G= (V, E) je Eulerjev sprehod. Pri tem sprehodu moramo prehoditi vsako povezavo natančno enkrat, lahko pa seveda obiščemo vozlišča večkrat. Utemeljite, zakaj ste pričeli in končali sprehod v vozlišču, v katerem ste ga začeli oziroma končali.

[1] Lahko predpostavite, da sta `who` in `time` celi števili – štartna številka in čas v sekundah.



Slika 1: Primer grafa.



Slika 2: Primer grafa.

4. naloga: Vračamo se h grafu na sl. 2, oziroma v splošnem h grafu $G(V, E)$, kjer $|V| = n$ in $|E| = m$.

VPRAŠANJA:

1. Spoznali smo dva algoritma za izgradnjo najcenejšega vpetega drevesa. (i) Katera sta ta dva algortima? (ii) Kakšna je časovna zahtevnost vsakega od njiju? (iii) V čem se algoritma razlikujeta?

2. V grafu sl. 2 poiščite najcenejše vpeto drevo. Pokažite izračun.

3. Recimo, da imajo v grafu vse uteži na povezavah težo 17. (i) Ali obstaja kakšen preprostejši algoritem za iskanje najcenejšega vpetega drevesa? Kakšna je njegova časovna zahtevnosti? Utemeljite odgovor.