

Konstrukcija i analiza algoritama - pismeni ispit JAN1

13.08.2025.

Ispit traje 3h. Svaki zadatak nosi po 10 poena. Ukupan broj poena na pismenom delu ispita je 60 poena. Prag za prolaz je 17 poena.

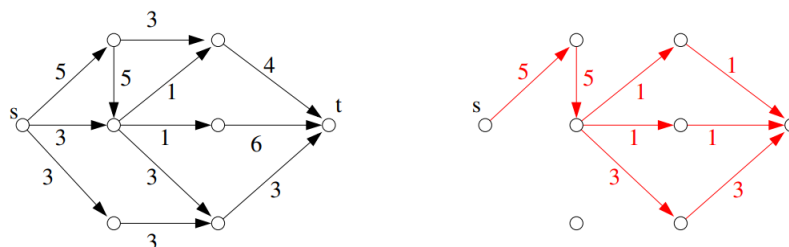
Problem obojivosti

Neka je $G = (V, E)$ neusmereni graf. Bojenje grafa podrazumeva da se iz skupa boja svakom čvoru dodeli tačno jedna boja. Problem 5-obojivosti je NP kompletni problem koji glasi: "Da li se proizvoljni graf može obojiti sa 5 boja tako da nikoja dva susedna čvora ne budu obojena istom bojom?". Za 5-bojenje kažemo da je pažljivo, ako nikoja dva susedna čvora nisu obojena istom bojom, ili bojama koje se razlikuju za 1 (mod 5). Preciznije, ako je skup boja $\{0, 1, 2, 3, 4\}$ dva susedna čvora ne mogu biti obojena bojama 0 i 4 ili bojama 2 i 3. Problem odlučivanja da li za dati graf postoji pažljivo 5-bojenje je takodje NP kompletni problem i to se može dokazati redukcijom problema 5-obojivosti na problem pažljive 5-obojivosti. Neka je G graf za koji želimo da donesemo odluku postojanja 5-obojivosti i neka je G' graf koji smo konstruisali tako što smo na svaku ivicu grafa G umetnuli po dva nova čvora (od ivice $u - v$ napravili smo put $u - x_{uv} - y_{uv} - v$).

- **(3 poena)** Ako je G 5-oboјivo, onda je G' pažljivo 5-oboјivo. Dokazati.
- **(3 poena)** Ako je G' pažljivo 5-oboјivo, onda je G 5-oboјivo. Dokazati.
- **(2 poena)** Data je staza od 4 čvora (graf oblika $a - b - c - d$). Nadji sve njene pažljive 5-oboјivosti. Koliko različitih pažljivih 5-oboјivosti postoji ako se boja prvog vrha fiksira na 0?
- **(2 poena)** Dat je ciklus C_5 (graf oblika $a - b - c - d - a$). Da li C_5 ima pažljivu 5-oboјivost? Ako da, napisi jedan primer, ako ne objasni zašto.

Maksimalni protok

Posmatrajmo transportnu mrežu (na slici levo) i jedan njen inicijalni tok f (na slici desno):



- **(2 poena)** Koji uslov treba da vazi kako bi za f mogli da kažemo da je tok? Kolika je vrednost toka f ?
- **(3 poena)** Krenuti od toka f i doći do maksimalnog toka. Objasniti postupak detaljno. Da li je f možda već maksimalan?
- **(3 poena)** Krenuti od trivijalnog toka (svuda 0 na početku) i doći do maksimalnog toka. Objasniti postupak detaljno.
- **(2 poena)** Minimum cut (minimalni rez) u transportnoj mreži je najmanji skup grana (ili čvorova) čije uklanjanje prekida svaki put od izvora (source, s) do odredišta (sink, t). Drugim rečima, "odsecanjem" tog skupa grana, nije moguće propustiti nijedan tok od s do t. Vrednost reza je suma kapaciteta svih grana koje idu iz "s strane" u "t stranu" reza. Minimum cut je onaj rez čija je vrednost najmanja moguća. Odrediti minimum cut za ovu transportnu mrežu.

Prefiksne sume na n -dimenzionalnoj hiperkocki

Posmatramo n -dimenzionalnu hiperkocku koja ima ukupno $P = 2^n$ procesora. Na početku, svaki procesor čuva tačno jedan element niza

$$A = (a_0, a_1, \dots, a_{P-1}),$$

pri čemu je indeks procesora jednak njegovoj n -bitnoj binarnoj adresi

$$(p_{n-1}p_{n-2} \dots p_1p_0)_2.$$

Za dati niz A , *prefiksna suma* na poziciji k definisana je kao

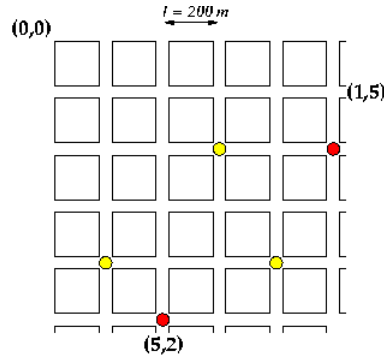
$$s_k = \sum_{i=0}^k a_i, \quad 0 \leq k \leq P-1.$$

Na kraju izvođenja algoritma, **svaki procesor** treba da zna vrednost prefiksne sume s_p koja se odnosi na njegov indeks p .

- **(2 poena)** Objasniti kako se adrese procesora predstavljaju u obliku binarnog koda i kako se pronalazi sused procesora u datoj dimenziji hiperkocke.
- **(2 poena)** Opisati kako se algoritam odvija kroz tačno n faza, uz obrazloženje zašto je toliko faza dovoljno da se izračunaju sve prefiksne sume.
- **(2 poena)** Precizno prikazati komunikacioni obrazac: koji procesor u svakoj fazi šalje podatke kom susedu, šta se tačno šalje, šta se prima i na osnovu kog kriterijuma procesor odlučuje da li primljenu vrednost treba dodati na svoju.
- **(2 poena)** Dati pseudokod ili jasan opis toka rada za jedan proizvoljan procesor (od inicijalne vrednosti do konačne prefiksne sume).
- **(2 poena)** Ilustrovati rad algoritma na primeru za $n = 3$: prikazati tabelu koja pokazuje vrednosti u svim procesorima nakon svake faze i objasniti kako su te vrednosti dobijene.

Taksi

U Novom Sadu sve ulice su izgrađene kao prave linije koje se seku pod pravim uglom na fiksnim rastojanjima, pri čemu je rastojanje između raskrsnica tačno 200 metara.



Jedan od glavnih događaja u Novom Sadu je Exit, i gotovo svi žele da dobiju kartu. Kako bi svi imali jednake šanse, mesto i vreme početka prodaje karata ostaju tajna sve do nekoliko minuta pre otvaranja šaltera. Kada se objavi otvaranje šaltera (putem Play radija), svi zainteresovani odmah pokušavaju da dodju do šaltera.

Najveću korist od toga ima vlasnik taksi službe. U trenutku objave putem radija, ljudi širom grada pozivaju centralu taksija tražeći prevoz. Problem za centralu je što mnogo ljudi traži prevoz u isto vreme i taksiji moraju veoma brzo da pakuju putnike.

Vaš zadatak je da pomognete centrali da odredi koliko maksimalno putnika može biti prevezeno uz sledeća ograničenja:

- Svaki taksi može da preveze samo jednog putnika.
- Putnici i taksiji čekaju na raskrsnicama.
- Taksi mora da stigne do putnika u datom vremenskom ograničenju.

Ulaz

Ulaz je u sledećem formatu:

- Jedna linija sa četiri broja: p (broj putnika, $1 \leq p \leq 400$), t (broj taksija, $1 \leq t \leq 200$), s (brzina taksija u m/s, $1 \leq s \leq 2000$), c (maksimalno vreme u sekundama, $1 \leq c \leq 10^6$).
- p linija sa koordinatama putnika (celobrojne koordinate).
- t linija sa koordinatama taksija (celobrojne koordinate).

Izlaz

Na standardni izlaz je potrebno ispisati maksimalan broj putnika koji mogu biti pokupljeni u datom vremenskom ograničenju.

Primer ulaza:

2 3 10 40

2 5

5 2

2 3

4 1

4 4

Izlaz:

2

Dinamična struktura

U ovom zadatku potrebno je održavati **strukturu celih brojeva** koja podržava sledeće dve osnovne operacije:

- **INSERT(S , x)** — ako broj x nije u strukturi S , umetni ga u S
- **DELETE(S , x)** — ako broj x postoji u strukturi S , izbriši ga

Pored toga, struktura mora da podržava i sledeće dve vrste upita:

- **K-TH(S)** — vrati k -ti najmanji element iz strukture S
- **COUNT(S , x)** — vrati broj elemenata u skupu S koji su strogo manji od x

Ulaz

Prva linija sadrži ceo broj Q ($1 \leq Q \leq 200\,000$) — broj operacija.

U narednih Q linija svaka linija sadrži po jednu operaciju, u sledećem formatu:

- karakter I , D , K ili C koji označava tip operacije:
 - I — umetanje (INSERT)
 - D — brisanje (DELETE)
 - K — upit za k -ti najmanji element (K-TH)
 - C — broj elemenata manjih od x (COUNT)
- zatim jedan ceo broj koji predstavlja parametar operacije

Za parametar x važi: $0 \leq |x| \leq 10^9$.

Za parametar k važi: $1 \leq k \leq 10^9$.

Izlaz

Za svaki upit K-TH i COUNT, ispisati rezultat u posebnoj liniji.

Za upit K-TH, ako je k veće od broja elemenata u skupu, ispisati reč - notvalid:

Primer ulaza:

```
8
I -1
I -1
I 2
C 0
K 2
D -1
K 1
K 2
```

Izlaz:

```
1
2
2
notvalid
```

Svi različiti podstringovi

Za dati string odštampati sve njegove različite podstringove. Podstringovi se najpre generišu i čuvaju u neuređenom skupu us , a potom se prolazi kroz skup i štampaju se podstringovi.

Primer ulaza:

aaabc

Izlaz:

bc b abc ab aabc aa aaa c a aaab aab aaabc