

Dizajn i analiza algoritama

Prvi domaći zadatak

MATEMATIČKI FAKULTET, UNIVERZITET U BEOGRADU
FILIP MARIĆ, MILAN MITRESKI

jesenji semestar 2025/2026

Tehnički detalji

Drage koleginice i kolege,

Pred vama se nalazi prvi domaći zadatak iz predmeta Dizajn i analiza algoritama u školskoj godini 2025/2026. Prvi domaći zadatak obuhvata teme koje smo obradjivali u prve tri nedelje nastave, na časovima vežbi: *Prefiskno drvo*, *Segmentno drvo i Union-Find*.

Domaći zadatak se sastoji od tri zadatka – po jedan iz svake teme. Broj poena koje je moguće ostvariti na svakom od zadataka dat je uz nazine zadataka. Maksimalan broj poena koji je moguće ostvariti na prvom domaćem zadatku je 5 poena.

Nije obavezno da uradite sve zadatake – možete uraditi jedan, dva ili sva tri zadatka (a možete da ne uradite nijedan, s obzirom da domaći zadaci nisu obavezni, ali pazite na ograničenje da je potrebno da na seminarском radu, kolokviju i domaćim zadacima ostvarite barem 25 poena).

Domaće zadatke predajete u vidu tri *cpp* datoteke (preporučujemo da konačna rešenja zadataka okačite na svoje Alas servere, radi lakše i efikasnije organizacije odbrane, ali ne tako da budu javno dostupni). Predaja i odbrana radova za prvi domaći zadatak organizovaće se u

petak, 12. decembra 2025. u 20:00 – sala 718

nakon časova predavanja. Odbranama prisustvuju profesor i asistent. Odbrane domaćih zadataka su obavezne. Ukoliko se na odbrani domaćeg zadatka ustanovi da kôd koji ste predali nije vaš i/ili ga ne razumete, gubite pravo na ostvarivanje poena preko domaćih zadataka.

Želimo vam puno sreće pri izradi domaćih zadataka!

Zadaci

Prefiksno drvo: Autocorrect [1.5 poen]

Tekst zadatka

Dat je niz stringova **words**, dužine n koji predstavlja rečnik poznatih reči. Korisnik sa standardnog ulaza unosi m reči iz rečnika i povremeno može pogrešiti. Za svaku unetu reč koja ne pripada rečniku, a u kojoj je korisnik pogrešio na **tačno jednom mestu**, ispisati spisak *svih mogućih reči* iz rečnika na koje je korisnik potencijalno mislio. Smatraći da korisnik uvek unosi reč sa jednakim brojem karaktera kao ispravna reč, tj. da samo može pogrešiti pri kucanju karaktera. Korisnik može uneti proizvoljnu reč - potrebno je detektovati isključivo reči koje sadrže jedan pogrešan karakter. U ostalim slučajevima ispisati prazan red.

Opis ulaza i izlaza

ULAZ: Na standardni ulaz unosi se broj n , a potom n stringova koji predstavljaju niz **words**. Nakon toga, unosi se broj m , a potom m reči.

IZLAZ: Na standardni izlaz za svaku od unetih m reči potrebno je ispisati sve reči (odvojene razmakom) u skladu sa uslovima zadatka.

Primer

ULAZ:

```
6
mikro
makro
mleko
red
sed
sistem
4
mekro
sleko
red
ped
```

IZLAZ:

```
makro mikro
mleko
sed red
```

Segmentno stablo: Dobre trojke [2 poena]

Tekst zadatka

Dat je niz pozitivnih celih brojeva **arr**. Uredjena trojka indeksa (i, j, k) za koju važi $i < j < k$ je *dobra trojka niza arr* ako važi jedan od naredna dva uslova: $\text{arr}[i] < \text{arr}[j] < \text{arr}[k]$ ili $\text{arr}[i] > \text{arr}[j] > \text{arr}[k]$. Odrediti broj uredjenih trojki niza **arr**.

Opis ulaza i izlaza

ULAZ: Na standardni ulaz unosi se broj n , a nakon toga n celih brojeva koji predstavljaju niz **arr**.

IZLAZ: Na standardni izlaz ispisati broj dobrih trojki niza **arr**.

Primer

ULAZ:

5
2 5 3 4 1

IZLAZ:

3

OBJAŠNJENJE: U pitanju su trojke $(0, 1, 3), (1, 2, 4)$ i $(1, 3, 4)$

Union-find: Relacije [1.5 poen]

Tekst zadatka

Dat je niz stringova `eq` koji predstavlja niz relacija izmedju promenljivih (mala slova abecede, u opsegu 'a'-'z') u obliku `x==y` ili `x!=y`. Potrebno je utvrditi da li je moguće dodeliti cele brojeve svim promenljivim tako da su sve zadate relacije ispunjene. Vrati `true` ako je to moguće, inače `false`.

Opis ulaza i izlaza

ULAZ: Na standardni ulaz unosi se broj n , a potom n stringova koji predstavljaju niz `eq`.

IZLAZ: Na standardni izlaz ispisati `true` ili `false` u skladu sa uslovima zadatka.

Primer 1

ULAZ:

```
2
a==b
b!=a
```

IZLAZ:

```
false
```

Primer 2

ULAZ:

```
5
a==b
c==d
b!=c
d==e
a!=e
```

IZLAZ:

```
true
```