

---

Elektrotehnički fakultet u Beogradu  
Katedra za računarsku tehniku i informatiku

*Predmet:* Algoritmi i strukture podataka 2 (13S112ASP2)  
*Nastavnici:* dr Milo Tomašević, red. prof.; doc. dr Marko Mišić;  
*Asistent:* Sanja Delčev, dipl. ing.; Maja Vukasović, dipl. ing.;  
Matija Dodović, dipl. ing.  
*Ispitni rok:* Drugi kolokvijum (decembar 2022. godine)  
*Datum:* 07.12.2022.

*Kandidat\*:* \_\_\_\_\_

*Broj Indeksa\*:* \_\_\_\_\_

*Kolokvijum traje 120 minuta, prvih sat vremena nije dozvoljeno napuštanje sale.  
Upotreba literature nije dozvoljena.*

<i>Zadatak 1</i>	_____ /10	<i>Zadatak 5</i>	_____ /15
<i>Zadatak 2</i>	_____ /15	<i>Zadatak 6</i>	_____ /10
<i>Zadatak 3</i>	_____ /15	<i>Zadatak 7</i>	_____ /10
<i>Zadatak 4</i>	_____ /10	<i>Zadatak 8</i>	_____ /15

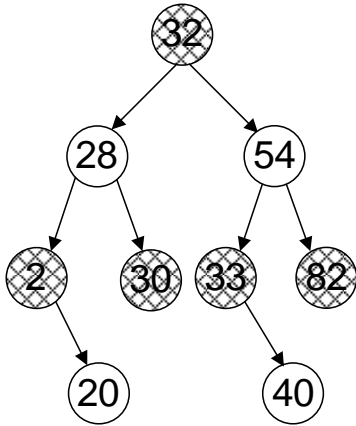
**Ukupno na kolokvijumu:** \_\_\_\_\_ /100

**Napomena:** Ukoliko u postavci nekog zadatka postoje nepreciznosti, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Na pitanja odgovarati **čitko, kratko i precizno**.

\* popunjava student.

1. [10] Izomorfizam crveno-crnih i 2-3-4 stabala.

Dato binarno crveno-crno stablo transformisati u izomorfno 2-3-4 stablo. Osenčeni čvorovi su crni. Iz dobijenog izomorfnog 2-3-4 stabla brišu se redom ključevi 33 i 82, zatim se dodaju ključevi 18, 24, 62 i 44, i na kraju se briše ključ 30. Nacrtati stablo nakon svake izmene.



2. [15] Posmatra se B\* stablo reda  $m$  ( $m > 3$ ) sa više od dva nivoa. Smatrati da čvor stabla sadrži sledeća polja: niz ključeva *keys*, niz pokazivača *children*, pokazivač na roditelja *parent* i broj ključeva u čvoru *num*. Poznato je sa su svi listovi minimalno popunjeni, a da unutrašnji čvorovi imaju više od minimalnog broja ključeva. Iz lista *node* briše se ključ *k*.
- a) [5] Koje su tri karakteristične situacije koje se mogu desiti prilikom ovog spajanja? Ukratko objasniti i po potrebi nacrtati delove stabla na kojima treba obeležiti čvor *node* iz kojeg se briše ključ.
- b) [10] Pored opisane strukture čvora dostupne su i sledeće funkcije koje se po potrebi mogu koristiti i koje nije potrebno implementirati: INDEX(*array*, *key*) koja vraća poziciju prvog ključa u nizu *array* koji je veći ili jednak *key* (od 1 do *num* ili 0 ako takav ključ ne postoji u čvoru); INSERT(*array*, *key*) koja u neopadajuće sortiran niz *array* ubacuje ključ *key*, i SHIFT(*node*, *i*) koja u čvoru B\* stabla *node* počevši od indeksa *i* pomera sve ključeve u nizu *keys* za jedno mesto ulevo i počevši od indeksa  $i+1$  pomera sve pokazivače u nizu *children* za jedno mesto ulevo. Napisati u pseudokodu funkciju koja realizuje opisano brisanje ključa *k* iz lista *node*.  
B-STAR-DELETE(*node*, *k*)

3. [15] Dato je *trie* stablo pokazivačem na koren stabla *root*, čiji su ključevi znakovni nizovi (mala slova engleskog alfabeta).

a) [5] Napisati u pseudokodu iterativnu implementaciju funkcije koja ispituje da li u stablu postoji ključ *key*.

HAS\_KEY(*root*, *key*)

b) [10] Napisati u pseudokodu funkciju koja ispituje da li se niz karaktera *sequence* može podeliti na ključeve koji se svi nalaze u stablu.

Dozvoljeno je koristiti rekurziju i pozivati funkciju implementiranu u tački a).

CAN\_BE\_DIVIDED(*root*, *sequence*)

4. **[10]** U inicijalno prazno B stablo reda 4 umeću se celobrojni ključevi 10, 17, 21, 9, 1, 12, 3, 8, 14, 4, 5. Nacrtati izgled stabla nakon svake izmene. Nakon umetanja svih ključeva izračunati popunjenost stabla, kao i prosečan broj pristupa prilikom uspešne i neuspešne pretrage.

5. [15] U nekom B+ stablu reda  $m$  čuvaju se celobrojni ključevi, koji ne moraju biti jedinstveni (moguće je čuvati i više stvarnih ključeva iste vrednosti).

a) [5] Objasniti na koji način bi bilo pogodno čuvati ključeve iste vrednosti, ukoliko je poznato da se uz same ključeve uvek čuvaju i neke korisne informacije.

b) [10] Za usvojenu strukturu pod a) implementirati funkciju `INSERT_DATA`, koja realizuje operaciju umetanja proizvoljnog ključa *key* i odgovarajućeg podatka *info* ukoliko je poznato da roditelj čvora u koji je umetanje izvršeno nije maksimalno popunjen. Smatrati da svaki čvor sadrži informaciju o trenutnoj popunjenosti čvora.

INSERT\_DATA (*root*, *m*, *key*, *info*)

6. [10] U neko digitalno stablo u implementaciji levi sin – desni brat redom umetnuti sledeće ključeve **malina, kutija, magija, mastilo, maska, kupina**, a potom ukloniti ključeve **maska** i **mastilo**. Prikazati stanje stabla nakon svih umetanja i, potom, nakon svih uklanjanja ključeva.
7. [10] Neka se posmatra jedno TOP-DOWN stablo  $m$ -arnog pretraživanja. Ukoliko je poznat broj ključeva  $n$  koji treba da bude smešten u stablu, kao i željena maksimalna visina stabla  $h$ , izvesti i objasniti izraz koji određuje stepen (red) stabla  $m$ . Prokomentarisati izbor vrednosti za  $m$  u odnosu na veličinu bloka podataka na disku, ukoliko se takvo stablo koristi za indeksiranje datoteke.

8. [15] Heširanje:

- a) [7] Na koji način se ključevi koji nisu numerički mogu transformisati u numeričke ključeve? Na primeru ključa  $azx$  koji pripada skupu ključeva koji se sastoje samo od malih slova engleskog alfabeta prikazati proces transformacije.

- b) [8] Neka se posmatra heš tabela sa 8 ulaza. Ukoliko se u nju smeštaju samo ključevi 3, 4, 5, 20, 21 sa diskretnom uniformnom funkcijom raspodele, navesti opšti oblik heš funkcije koja se može iskoristiti i prikazati proces popunjavanja tabele. Koja je prednost ovog metoda u odnosu na izbor nezavisne heš funkcije?

0	
1	
2	
3	
4	
5	
6	
7	