
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Algoritmi i strukture podataka (13E112ASP)
Nastavnici: dr Milo Tomašević, red. prof.; doc. dr Marko Mišić
Asistenti: Sanja Delčev, dipl. ing.; Maja Vukasović, dipl.ing.;
Milica Despotović, dipl.ing.
Ispitni rok: Januar 2020.
Datum: 19.01.2020.

Kandidat:* _____

Broj Indeksa:* _____

Prvi deo ispita traje 120 minuta. Drugi deo ispita traje 30 minuta.

Studenti koji žele da im se prizna domaći zadatak umesto drugog dela ispita treba to da naznače na prvoj stranici.

Napuštanje sale nije dozvoljeno tokom prvih 60 minuta.

Upotreba literature nije dozvoljena.

Zadatak 1 _____ /15

Zadatak 4 _____ /15

Zadatak 2 _____ /15

Zadatak 5 _____ /10

Zadatak 3 _____ /15

Zadatak 6 _____ /10

Prvi deo ispita: _____/80

Drugi deo ispita: _____/20

Ukupno na ispitu: _____/100

Napomena: Ukoliko u postavci nekog zadatka postoje nepreciznosti, student treba da uvede razumnu pretpostavku, da je u okviru (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Kod pitanja koja imaju ponuđene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

* popunjava student.

Prvi deo ispita (strane 2 - 6)

1. [15] Posmatra se rastuće uređeni **binarni hip**, koji je dat u vidu niza ključeva *heap*. Napisati u pseudokodu efikasnu iterativnu funkciju koja menja vrednost zadatog ključa *key*, na zadatu vrednost *new_key*. Poznato je da hip ima *n* elemenata.

HEAP CHANGE(*heap, n, key, new key*)

2. [15] Podaci se smeštaju u heš tabelu sa 7 ulaza primenom heš funkcije $h_p(K) = K \bmod 7$. Za razrešavanje kolizija se koristi tehnika kvadratnog pretraživanja. U tabelu se umeću redom ključevi 38, 31, 10, 56, 21. Odrediti prosečan broj pristupa prilikom uspešnog i neuspešnog traženja i verovatnoću popunjavanja praznih ulaza, pod pretpostavkom da su svi ključevi jednako verovatni.

0	
1	
2	
3	
4	
5	
6	

3. [15] U inicijalno prazno B stablo se redom umeću ključevi 16, 54, 42, 72, 4, 36, 62 i 9. Koliki je prosečan broj pristupa pri uspešnoj pretrazi nakon umetanja svih ključeva? Iz ovako formiranog stabla se zatim uklanjaju ključevi 62, 16 i 42. Pri uklanjanju se koristi **prethodnik**. Prikazati dodavanje i uklanjanje ključeva po koracima.

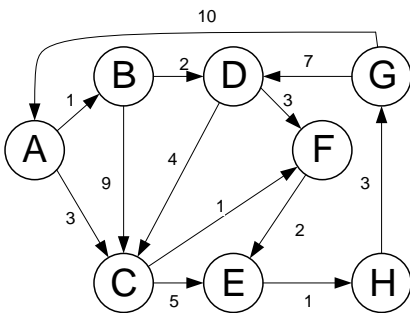
4. [15] Da bi olakšao učenje student je čitajući knjigu obeležio stikerima stranice koje odvajaju veće celine u knjizi. Svaka veća celina sastoji se od određenog broja stranica.
- a) [6] Predložiti strukturu kojom se efikasno modeluje ovako obeležena knjiga. Smatrati da se jednom postavljeni stikeri ne pomeraju.

- b) [9] Implementirati funkciju `FIND_PAGE` koja treba da u skladu sa usvojenom strukturom pronađe odgovarajuću stranicu.

`FIND PAGE(book, page)`

5. [10] Data je sekvenca ključeva: 159, 57, 103, 7, 74, 95, 8, 101, 179, 45, 303, 42, 219. Prikazati rad *shell sort* algoritma po koracima. Navesti korišćenu sekvencu inkremenata i obrazložiti izbor.

6. [10] Korišćenjem Dijkstra algoritma, odrediti najkraća rastojanja od čvora A do svih ostalih čvorova za graf sa slike. Rad algoritma prikazati po koracima.



Drugi deo ispita – programski zadatak (strane 7 - 8)

7. [20] Napisati efikasnu **iterativnu** implementaciju funkcije na programskom jeziku C/C++ koja prihvata pokazivač na koren stabla binarnog pretraživanja *root* i treba da ažurira stablo tako što će svaki ključ u njemu zameniti sumom svih ključeva većih od njega. Nije dozvoljeno koristiti gotove strukture podataka.

```
struct Node {
    struct Node* left;
    struct Node* right;
    int          value;
};

void replace_with_greater_sum(Node* root);
```

