
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
Asistenti: Sanja Delčev, dipl. ing.; Maja Vukasović, dipl.ing.;
Dragana Milovančević, dipl.ing.
Ispitni rok: Treći kolokvijum (januar 2019. godine)
Datum: 20.01.2019.

Kandidat:* _____

Broj Indeksa:* _____

*Kolokvijum traje 100 minuta. Napuštanje sale nije dozvoljeno tokom prvih 60 minuta.
Upotreba literature nije dozvoljena.*

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

Ukupno na kolokvijumu: _____ /100

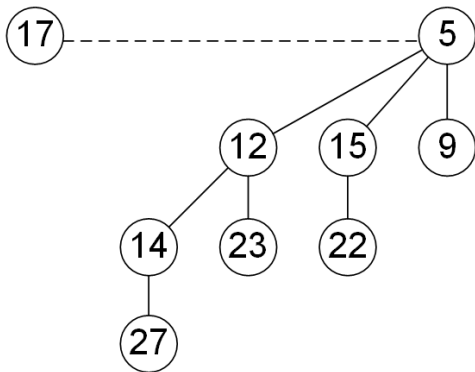
Napomena: Ukoliko u postavci nekog zadatka postoje nepreciznosti, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Kod pitanja koja imaju ponuđene odgovore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

* popunjava student.

1. [15] Upotrebom metode heširanja implementirati funkciju `FIND_NUM_OF_PAIRS` koja treba da pronađe ukupan broj parova celobrojnih vrednosti u zadatom nizu `arr` koji u sumi po modulu n daju vrednost k koja predstavlja parametar funkcije. Za realizaciju ove funkcije treba koristiti metodu odvojenog ulančavanja.

`FIND_NUM_OF_PAIRS(arr, n, k)`

2. [10] Prikazati izgled **binomnog hipa** nakon svake izmene, ukoliko se umeću ključevi 19, 11, 25 i nakon toga obriše minimum.



3. [10] Data je sekvenca ključeva: 27, 25, 14, 34, 7, 1, 5, 193, 33, 41, 73, 124. Prikazati rad *radix sort* algoritma po koracima.

4. [10] Podaci se smeštaju u heš tabelu sa 10 ulaza primenom heš funkcije $h_p(K) = K \bmod 10$. Za razrešavanje kolizija se koristi metoda objedinjenog ulančavanja. Prikazati popunjavanje date tabele nakon umetanja svakog od sledećih ključeva: 37, 41, 18, 16, 12. Nakon svih umetanja, odrediti verovatnoću popunjavanja praznih ulaza, pod pretpostavkom da su svi ključevi jednako verovatni.

0	1	2	3	4	5	6	7	8	9
	11							21	29
-1	8	-1	-1	-1	-1	-1	-1	-1	-1

free

5. [15] Priloženi pseudokod prikazuje algoritam za uređivanje špila karata u jednoj kartaškoj igri. U igri se koristi špil od po 8 karata za svaki od 4 znaka (ukupno 32). Algoritam treba da grupiše karte istog znaka, pri čemu karte istog znaka sortira prema rastućoj vrednosti. Uočiti i objasniti greške ili nedostatke priloženog algoritma i napisati u pseudokodu ispravnu implementaciju u skladu sa zahtevima. Polje *val* predstavlja vrednost karte, a polje *sign* njen znak.

```
for i = 2 to 32 do
  k = cards[i]
  j = i-1
  while(j>0 and cards[j].val>k.val) do
    cards[j+1] = cards[j]
    j = j-1
  end_while
  cards[j] = k
end_for
for i = 1 to 4 do
  c[i] = 0
end_for
for j = 1 to n do
  c[cards[j].sign] = c[cards[j].sign]+1
end_for
for i = 3 downto 1 do
  c[i] = c[i+1] - c[i]
end_for
for j = 1 to n do
  b[c[cards[j].sign]] = cards[j]
  c[cards[j].sign] = c[cards[j].sign]-1
end_for
```

SORT(cards)

6. [15] Koristeći ideju *quick sort* algoritma, napisati u pseudokodu funkciju koja vraća k -ti najmanji element u zadanom nizu celih brojeva arr . Prilikom izbora *pivot*-a, koristiti medijanu od tri pogodno izabrane vrednosti.

QUICK_SELECT(arr, k)

7. [10] Podaci se smeštaju u heš tabelu sa 9 ulaza primenom heš funkcije $h_p(K)=K \bmod 9$. Za razrešavanje kolizija se koristi metoda linearnog pretraživanja sa korakom 2. Definisati pojam primarnog i sekundarnog grupisanja, a zatim na primeru delimično popunjene tabele sa slike, ilustrovati jedan scenario umetanja kod kojeg se dešava primarno grupisanje i jedan scenario umetanja kod kojeg se dešava sekundarno grupisanje.

0	27
1	
2	11
3	
4	
5	23
6	
7	
8	8

8. [15] Neka se kod spoljašnjeg heširanja koristi heš tabela sa 4 baketa, koji imaju kapacitet $b = 3$. Za heš funkciju se koristi metod deljenja $K \bmod 4$, a za razrešavanje kolizije linearno pretraživanje. Neka se pri brisanju koristi metod selektivnog pomeranja (analogno kao kod unutrašnjeg heširanja). U tabelu treba umetnuti ključeve 60, 18, 23, 50, 47, 34, 82, 14, 67 i 38, a zatim brisati ključeve 50, 47, 23 i 34. Prikazati izgled tabele nakon svakog razrešenja kolizije pri umetanju, kao nakon svakog brisanja.