

Primer pitanja za teorijski test iz Konstrukcije i analize algoritama, I smer, 2023/24.

KOREKTNOST

1. Definisati pojam invarijante petlje.
2. Navesti šta je invarijanta spoljašnje petlje algoritma sortiranja umetanjem (Insertion sort).
3. Navesti šta je invarijanta algoritma binarne pretrage.
4. Koliki je otprilike opseg brojeva koji se mogu smestiti u promenljivu tipa int (32 bita), a koliki u promenljivu tipa long long (64 bita)?

SLOŽENOST

5. Kolika je suma vrednosti 2^k za $k=0$ do n ?
6. Kolika je suma vrednosti $(1/2)^k$ za $k=0$ do n ?
7. Kojom brzinom asimptotski u beskonačnost raste suma $1^2 + 2^2 + \dots + n^2$?
8. Šta je harmonijski red? Da li je on konvergentan? Kojom brzinom rastu njegove parcijalne sume?
9. Reši rekurentnu jednačinu $T(n) = 2T(n-1) + O(1)$, $T(0) = O(1)$.
10. Reši rekurentnu jednačinu $T(n) = 3T(n/3) + O(n)$, $T(0) = O(1)$.
11. Reši rekurentnu jednačinu $T(n) = 5T(n/5) + O(1)$, $T(0) = O(1)$.
12. Reši rekurentnu jednačinu $T(n) = 3T(n/2) + O(n^2)$, $T(0) = O(1)$.
13. Navedi bar dva algoritma tipa podeli pa vladaj koji imaju različite jednačine kojima se opisuje njihova složenost. Postavi i reši te jednačine.
14. Navesti bar dva primera algoritama koji imaju rekurentnu jednačinu $T(n) = 2T(n/2) + O(n)$, $T(0) = O(1)$. Koja je njihova složenost?
15. Navesti primer petlje for koja se izvršava u a) linearnoj b) logaritamskoj c) konstantnoj vremenskoj složenosti.
16. Navesti primer algoritma složenosti $O(\log n)$ i primer algoritma složenosti $O(\sqrt{n})$. Koji se od njih smatra efikasnijim? Koliki je otprilike red veličine broja n da bi se algoritam mogao sprovesti za oko jedne sekunde za svaku od ove dve složenosti?
17. Koliki je otprilike maksimalni red veličine ulaza n koji se može obrađivati algoritmima složenosti 2^n ?
18. Koliki je otprilike red veličine ulaza koji se na prosečnom današnjem računaru može rešiti algoritmom kvadratne složenosti za jednu sekundu?
19. Koliko operacija sabiranja celih brojeva otprilike može da izvrši prosečan današnji računar u toku jedne sekunde?
20. Koliko otprilike operacija izvršava algoritam složenosti $n \log n$ za ulaz veličine $n=10^6$?

SORTIRANJE

21. Koje vreme zahtevaju efikasni algoritmi za sortiranje niza od n elemenata? Koje su dve operacije osnovne tokom sortiranja i koliko smatramo da je njihova složenost?
22. Opisati bar dva načina da se efikasno (u kvazilinearnom vremenu) izbroje svi različiti elementi date serije brojeva.
23. Opisati bar jedan efikasan način particionisanja koji se može koristiti u brzom sortiranju. Koja je složenost koraka particionisanja?
24. Kako se bira pivot u algoritmu brzog sortiranja?

25. Kojom rekurentnom jednačinom se može opisati vreme izvršavanja algoritma MergeSort?

REKURZIJA I DP

26. Kakav je odnos veličine raspoložive stek memorije i hip memorije na današnjim računarima? Koja se zona memorije koristi za realizaciju rekurzije?
27. Eliminirati rekurziju iz funkcije `void f(n) { cout << 3 << endl; if (n > 0) f(n-1); }`
28. Navesti primer problema preklapanja rekurzivnih poziva (ponovljenih rekurzivnih poziva) i opisati kako se on rešava.
29. Da li se prilikom tehnike memoizacije eliminiše rekurzija?
30. Opiši algoritam za nabranje svih varijacija sa ponavljanjem dužine n skupa {1, ..., k}.

OSNOVNE STRUKTURE PODATAKA

31. Ako su poznati pokazivači na početak i na kraj jednostruko povezane liste, koja je složenost operacija dodavanja i brisanja sa početka i sa kraja liste.
32. Kako je implementirana neuređena mapa, a kako uređena mapa?
33. Navesti bar tri operacije koje se efikasno mogu izvršiti nad uređenim skupom, a ne mogu nad neuređenim.
34. Koje su osnovne operacije koje podržava tip podataka skup?
35. Koja je složenost provere da li se element nalazi u skupu ako je skup implementiran preko a) pretraživačkog drveta koje ne mora biti balansirano b) pretraživačkog drveta koje je uvek balansirano?
36. Koja su dva osnovna načina implementacije asocijativnih struktura podataka (skupova i mapa)?
37. Kada se koristi heš-tabela kako se dobija pozicija u tabeli na kojoj bi trebalo da se smesti dati element?
38. Šta je kolizija heš-funkcije? Opisati neki način razrešavanja kolizija.
39. Koja je složenost operacija dodavanja i brisanja elemenata sa početka i sa kraja i koja je složenost indeksnog pristupa kod reda sa dva kraja ako je on implementiran preko a) dvostruko povezane liste i b) preko strukture podataka dek (engl. deque)?
40. Koje su osnovne tri operacije reda sa prioriteto?
41. Kako se u memoriji predstavlja struktura podataka hip?
42. Koji je odnos roditelja i njegova dva deteta u min-hipu?
43. Koji je odnos roditelja i njegova dva deteta u pretraživačkom binarnom drvetu?

NAPREDNE STRUKTURE PODATAKA

44. Složenost pretrage reči u prefiksnom drvetu zavisi od (zaokružiti sve tačne odgovore): a) ukupnog broja reči u drvetu b) dužine reči koja se traži c) dužine najduže reči u rečniku
45. Opiši dve osnovne operacije strukture union-find tj. disjoint set union.
46. Kako se može preprocesirati niz tako da se nakon toga u konstantnom vremenu mogu izračunavati zbirni segmenta uzastopnih elemenata?
47. Ako su poznate razlike susednih elemenata niza, kako se može rekonstruisati taj niz?

48. Kako se može pretprocesirati niz tako da se nakon toga u konstantnom vremenu mogu ažurirati svi elementi segmenta uzastopnih elemenata (uvećavanjem svih elemenata segmenta za istu vrednost)?
49. Uporediti segmentna i Fenwickova drveća: a) primenljiva su na isti skup operacija b) segmentna se mogu primeniti na širi skup operacija c) Fenwickova se mogu primeniti na širi skup operacija d) postoje operacije koje je moguće obraditi samo Fenwickovim i operacije koje moguće rešiti samo segmentima drvetom
50. Koji upiti se mogu realizovati u složenosti $O(\log n)$ kod segmentnih drveća: a) očitavanje vrednosti jednog elementa b) ažuriranje jednog elementa c) ažuriranje segmenta d) računanje zbira elemenata segmenta
51. Koja se vrsta upita dodatno omogućava korišćenjem lenjih segmentnih drveća?

GRAFOVI

52. Kakva je matrica susedstva neusmerenih grafova?
53. Navesti bar dve operacije koje je efikasnije uraditi u reprezentaciji grafa matricom povezanosti nego listama povezanosti.
54. Koliko grana ima usmeren kompletni graf (bez petlji) nad 6 čvorova?
55. Koliko grana ima neusmeren kompletni graf (bez petlji) nad 6 čvorova?
56. Koji tipovi grana postoje u odnosu na DFS pretragu kod neusmerenih grafova?
57. Koji tipovi grana postoje u odnosu na DFS pretragu kod usmerenih grafova?
58. Koji je dolazni, a koji odlazni redni broj čvora iz koga pokrećemo pretragu povezanog neusmerenog grafa u dubinu?
59. Za koje tipove grana (u, v) u usmerenom grafu važi odnos $u.Pre > v.Pre$ (gde su $u.Pre$ i $v.Pre$ brojevi u dolaznoj numeraciji)? a) grane drveća b) povratne grane c) poprečne grane d) direktne grane
60. Da li za neki tip grane (u, v) može istovremeno da važi i $u.Pre < v.Pre$ i $u.Post < v.Post$?
61. Koliki je maksimalni broj elemenata u redu tokom BFS obilaska?
62. Koji od navedenih algoritama su složenosti $O(V+E)$? a) BFS b) DFS c) Kanov algoritam d) Dajkstrin algoritam e) Tardžanov algoritam za jake komponente
63. Nacrtati primer grafa sa 6 čvorova i tačno 3 komponente jake povezanosti.
64. Da li usmeren aciklički graf može da ima više različitih topoloških uredjenja? Da li za usmereni aciklički graf može da ne postoji topološko uredjenje?
65. Prilikom implementacije Kanovog algoritma čvorovi stepena nula se smeštaju a) isključivo u red b) isključivo na stek c) svejedno je da li se koriste red ili stek.
66. Prilikom implementacije Tardžanovog algoritma za komponente jake povezanosti elementi se smeštaju a) isključivo u red b) isključivo na stek c) svejedno je da li se koristi red ili stek
67. Da li svi čvorovi/nijedan čvor grafa mogu biti artikulacione tačke?
68. Da li sve grane/nijedna grana u grafu mogu biti mostovi?
69. Nacrtati graf od 5 čvorova koji ne sadrži nijedan most.
70. Kada je vrednost *low link* čvora u neusmerenom grafu jednaka njegovom rednom broju u dolaznoj numeraciji?

71. Ako iz čvora v neusmerenog grafa čiji je redni broj u dolaznoj numeraciji 10 postoje grane ka čvorovima sa rednim brojevima 2, 5, 9, 11 i 13, koja će biti njegova *low link* vrednost?
72. Kojim redosledom se obrađuju čvorovi u algoritmu za traženje najkraćih puteva od zadanog čvora u acikličkim grafovima?
73. U kom redosledu se obrađuju čvorovi u Dajkstrinom algoritmu?
74. Koja struktura podataka se koristi kod efikasne implementacije Primovog algoritma?
75. U kom grafovskom algoritmu se koristi union-find struktura podataka?
76. Kojim redosledom se obrađuju grane u Kraskelovom algoritmu?
77. Kada težinski graf može da sadrži više različitih minimalnih povezujućih drveta?
78. Ojlerov ciklus sadrži tačno jednom a) svaki čvor b) svaku granu grafa
79. Da li je teže ispitati da li graf sadrži Ojlerov ciklus ili Hamiltonov ciklus?
80. Koji algoritam se može primeniti na grafove koji imaju negativne težine grana? a) Dajkstrin c) Belman-Fordov d) Flojd-Varšalov
81. Indukcija kod Flojd-Varšalovog algoritma teče po: a) broju čvorova grafa b) broju grana grafa c) skupu čvorova koji se mogu javiti na putevima između dva krajnja čvora.
82. Flojd-Varšalov algoritam je primer koje strategije: a) pohlepnih algoritama b) dinamičkog programiranja c) dekompozicije
83. Koji se algoritam može koristiti za određivanje tranzitivnog zatvorenja relacije?
84. Šta treba da važi za relaksaciju grana na najkraćem putu od čvora u do čvora v da bi taj najkraći put bio određen?

ALGEBRA I TEORIJA BROJEVA

85. Koja je složenost klasičnog algoritma provere da li je broj prost?
86. Koja je složenost Eratostenovog sita?
87. Koja je složenost Euklidovog algoritma zasnovanog na oduzimanju, a koja Euklidovog algoritma zasnovanog na deljenju?
88. Kada se mogu pronaći celi brojevi x i y tako da je $ax+by=c$?
89. Koja je složenost najgoreg slučaja faktorizacije broja n osnovnim algoritmom (proverom delilaca)? Da li se taj algoritam efikasno može sprovesti za podatke tipa int ? Da li se taj algoritam može efikasno sprovesti za brojeve zapisane sa 4096 bitova? Da li se može sprovesti bar za neki broj zapisan sa 4096 bitova?
90. Kada za funkciju kazemo da je multiplikativna?
91. Navesti bar dve multiplikativne funkcije.
92. Da li je lakše odrediti vrednost Ojlerove funkcije broja ili odrediti nzd dva broja?
93. Šta je grupa? Šta je modularna multiplikativna grupa?
94. Kada postoji modularni multiplikativni inverz broja x po modulu m ?
95. Da li je lakše odrediti modularni multiplikativni inverz broja ili faktorisati broj?
96. Da li je efikasniji algoritam za računanje modularnog multiplikativnog inverza zasnovan na Ojlerovoj teoremi ili na proširenom Euklidovom algoritmu?
97. Koji su uzajamno prosti brojevi sa brojem 9? Koliko ih ima? Koji je ostatak pri deljenju broja 4^6 sa 9?
98. Koja je složenost algoritma za brzo stepenovanje broja: a^k ?

99. Da li se u razumnom vremenu može odrediti proizvod dva broja po datom modulu, ako su ti brojevi i modul zapisani sa po oko 1000 cifara? Da li se za dati proizvod po modulu i jedan činilac u razumnom vremenu može odrediti drugi činilac?
100. Da li se u nekom razumnom vremenu može izračunati stepen po modulu ako su osnova, eksponent i moduo brojevi zapisani sa po oko 1000 cifara? Da li se za datu vrednost stepena može u razumnom vremenu odrediti eksponent?
101. Naći broj koji pri deljenju sa 2 daje ostatak 1, sa 3 ostatak 2, a sa 5 ostatak 4. Koja teorema garantuje njegovo postojanje i daje način da se taj broj izračuna?

OBRADA TEKSTA

102. Kolizija heš-funkcije je u vezi sa njenom (a) surjektivnošću (b) injektivnošću (c) parnošću (d) neparnošću
103. Koje je složenosti upoređivanje dve niske dužina redom n_1 i n_2 , ako koristimo heširanje i imamo izračunate njihove heš vrednosti? Koje je složenosti upoređivanje dve niske dužina redom n_1 i n_2 , ako koristimo heširanje i tek treba da izračunamo njihove heš vrednosti?
104. Rabin-Karpov algoritam se koristi za (a) određivanje heš vrednosti proizvoljnog segmenta niske (b) traženje niske u tekstu (c) računanje najdužih prefiks-sufiksa prefiksa niske
105. Tehnika kojom se postiže efiksanost u Rabin-Karpovom algoritmu je (a) sortiranje (b) odsecanje (c) inkrementalnost (d) binarna pretraga
106. Ako je poznata heš-vrednost podniske $s_i..s_{i+n-1}$, u kom vremenu se može izračunati heš-vrednost podniske $s_{i+1}...s_{i+n}$?
107. Šta čuva z-niz? (a) dužinu najdužeg prefiks-sufiksa prefiksa niske do tog karaktera (b) dužinu najdužeg segmenta niske sa početkom na toj poziciji koji se poklapa sa prefiksom niske (c) dužinu najdužeg palindroma prefiksa niske do tog karaktera
108. Kako bi izgledao z-niz niske abcxabyab?
109. Kada se ažurira z-kutija u z-algoritmu? (a) kada se naiđe na z-kutiju sa većom levom granicom (b) kada se naiđe na z-kutiju sa većom desnom granicom (c) kada se naiđe na dužu z-kutiju
110. Koja je složenost popunjavanja z-niza korišćenjem z-algoritma?
111. Koja je složenost naivnog algoritma za pretragu teksta u najgorem slučaju, ako je n dužina teksta, a m dužina niske? (a) $O(n*m)$ (b) $O(\log n)$ (c) $O(n+m)$ (d) $O(n \log n)$
112. Koja je složenost KMP algoritma u najgorem slučaju, ako je n dužina teksta, a m dužina niske? (a) $O(n*m)$ (b) $O(\log n)$ (c) $O(n+m)$ (d) $O(n \log n)$
113. Navesti sve prefiks-sufikse niske ababab
114. Izračunati prefiksnu funkciju koja se koristi u KMP algoritmu za nisku ABCABDCDBA.
115. Koja je složenost preprocesiranja niske dužine m u KMP algoritmu?
116. Koji od navedenih algoritama za traženje niske u tekstu su u mestu (zahtevaju dodatni memorijski prostor veličine $O(1)$) (a) naivni algoritam (b) KMP algoritam (c) algoritam zasnovan na z-nizu
117. Opisati opšti oblik niske za koju je vrednost $\sum_{i=1}^n pi[i]$ maksimalna

GEOMETRIJA

118. Koji od navedenih problema se mogu rešiti korišćenjem skalarnog proizvoda? (a) izračunavanje konveksnog ugla između vektora (b) površina trougla (c) intenzitet vektora (d) orijentacija trojke tačaka (e) kolinearnost tačaka
119. Koji od navedenih problema se mogu rešiti korišćenjem vektorskog proizvoda? (a) izračunavanje konveksnog ugla između vektora (b) površina trougla (c) intenzitet vektora (d) orijentacija trojke tačaka (e) kolinearnost tačaka
120. Koji od narednih izraza je jednak $a \times b$, gde su a i b dva vektora?
- $-(a \times b)$
 - $a \circ b$
 - $b \times a$
 - $-(b \times a)$
121. Kojim od narednih izraza se dobija vektor upravan na vektore a i b ?
- $a \times b$
 - $a \circ b$
 - $b \times a$
 - $i a \times b$ i $b \times a$
122. Dva vektora su kolinearna ako i samo ako im je (a) skalarni proizvod jednak nuli (b) vektorski proizvod jednak nuli (c) skalarni proizvod jednak 1 (d) skalarni proizvod jednak -1
123. Dužina normalne projekcije vektora na pravac drugog vektora se može izračunati korišćenjem: (a) zbira ta dva vektora (b) skalarnog proizvoda ta dva vektora (c) vektorskog proizvoda ta dva vektora (d) mešovitog proizvoda vektora
124. Vektorski proizvod dva vektora je: (a) upravan na tačno jedan od vektora koji učestvuju u vektorskom proizvodu (b) upravan na ravan koja sadrži oba vektora (c) paralelan sa tačno jednim od vektora koji učestvuju u vektorskom proizvodu (d) paralelan sa ravni koja sadrži oba vektora
125. Za šta se koristi formula pertle?
126. U algoritmu za određivanje preseka horizontalnih i vertikalnih duži se u skupu kandidata čuvaju horizontalne duži za koje važi:
- njihov i levi i desni kraj su levo od tekućeg položaja pokretne prave
 - njihov i levi i desni kraj su desno od tekućeg položaja pokretne prave
 - njihov levi kraj je levo, a desni desno od tekućeg položaja pokretne prave
127. Za kakav mnogougao kažemo da je prost?
128. Za kakav mnogougao kažemo da je konveksan?
129. Koja je složenost algoritma za utvrđivanje da li je tačka u prostom mnogouglu?
130. Koja je složenost algoritma za utvrđivanje da li je tačka u konveksnom mnogouglu?
131. Da li za dati skup od tri ili više nekolinearnih tačaka uvek postoji prost mnogougao koji ih povezuje?
132. Koja je složenost algoritma za konstrukciju prostog mnogougla?
133. Da li je prost mnogougao nad datim skupom tačaka jedinstven? Da li je konveksni omotač datog skupa tačaka jedinstven?

134. Da li se temena tekućeg konveksnog omotača dobijenog nakon k koraka nužno nalaze i u konačnom konveksnom omotaču u (a) direktnom induktivnom algoritmu (b) algoritmu uvijanje poklona (c) Grejemovom algoritmu
135. Koja je prosečna složenost brzog algoritma za konstrukciju konveksnog omotača?
136. Koji od navedenih algoritama za konstrukciju konveksnog omotača su u najgorem slučaju složenosti $O(n \log n)$: (a) inkrementalni algoritam (b) algoritam uvijanja poklona (c) Grejemov algoritam (d) brzi algoritam
137. Koja struktura podataka se može koristiti u Grejemovom algoritmu? (a) red (b) stek (c) sortirana lista (d) red sa prioritetom