

Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Algoritmi i strukture podataka 1 (13S111ASP1)

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: Prvi kolokvijum (mart 2023.)

Datum: 27.03.2023.

Kandidat^{*}: _____

Broj Indeksa^{*}: _____

*Kolokvijum traje 90 minuta. Prvih 60 minuta od početka nije dozvoljeno napuštanje sale.
Upotreba literature nije dozvoljena.*

Zadatak 1	_____ /15	Zadatak 4	_____ /15
Zadatak 2	_____ /20	Zadatak 5	_____ /15
Zadatak 3	_____ /15	Zadatak 6	_____ /20

Ukupno na kolokvijumu: _____ /100

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

* popunjava student.

1. [15] Data je retko popunjena kvadratna matrica $A[0:N-1, 0:N-1]$. Nepodrazumevani elementi matrice smeštaju se u memoriju po vrstama, a matrica se može posmatrati kao donje trougaona u odnosu na sporednu dijagonalu, kao što je prikazano na slici.

a) [10] Izvesti adresnu funkciju za pristup proizvoljnom elementu matrice. Jasno naznačiti uslove za pristup podrazumevanim i nepodrazumevanim elementima.

				X
			X	X
		X	X	X
	X	X	X	X
X	X	X	X	X

b) [5] Implementirati funkciju GET koja vraća vrednost traženog elementa sa indeksima i i j .

GET(A, i, j, N)

2. [20] Neka se posmatra jedan XML dokument koji može sadržati tagove oblika `<tag>`. Određeni tagovi dolaze u obliku otvorenog i zatvorenog taga (oblika `</tag>`) koji se mogu ugneždavati, a neki ne. Napisati u pseudokodu funkciju koja proverava da li je sekvenca tagova korektno ugneždena. Na raspolaganju je funkcija `IS_NESTABLE` koja vraća *true* ukoliko tag pripada grupi koja se može ugneždavati, a *false* u suprotnom. Potrebne linearne strukture podataka implementirati korišćenjem sekvensijalne reprezentacije.

CHECK TAGS(*code*)

3. [15] Prikazati po koracima konverziju datog izraza iz infiksne u postfiksnu notaciju i dopuniti tabelu prioriteta. Operator \wedge označava stepenovanje, a $=$ predstavlja dodelu vrednosti sa najmanjim prioritetom i smerom grupisanja s desna na levo. Smatrati da preostali operatori imaju standardne odnose prioriteta i smerove grupisanja i na osnovu toga popuniti priloženu tabelu.

$$E = (B * (A = B + C)) - (A - D / E)$$

Operatori	IPR	SPR
\wedge		
$+, -$		
$*, /$		
$=$		
(
)		

Input	Stack	Postfix
E		
=		
(
B		
*		
(
A		
=		
B		
+		
C		
)		
)		
-		
(
A		
-		
D		
/		
E		
)		
EOF		

4. [15] Objasniti način smeštanja retkih matrica u *Compressed Storage Row* (CSR) formatu i na koji način se može odrediti broj elemenata u svakoj vrsti i koloni na osnovu ovog formata. Retko posednutu matricu sa slike prikazati u navedenom formatu.

	4	2	5
		8	
	13	15	
		4	6
	11		

5. [15] Neka je neprioritetni red implementiran tehnikom kružnog bafera u vektoru $Q[1:6]$. Navesti formule za održavanje pokazivača *front* i *rear* u konkretnom slučaju, a zatim prikazati stanje reda po koracima nakon umetanja vrednosti 5, 3, 8, 13, brisanja dve vrednosti, umetanja 7, 16, 5 i dohvatanja tri vrednosti. Obeležiti na slici i jasno navesti vrednosti pokazivača *front* i *rear* nakon svake sekvence umetanja i brisanja.

6. [20] Neka je data jedna jednostruko ulančana nekružna lista celih brojeva na koju ukazuje pokazivač *list*. Napisati u pseudokodu implementaciju funkcije koja na što **efikasniji** način zamenjuje mesta prvoj i drugoj polovini liste. U slučaju da lista ima neparan broj elemenata, smatrati da se u prvoj polovini nalaze svi elementi pre središnjeg elementa u listi.

Primer: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ se menja u $4 \rightarrow 5 \rightarrow 6 \rightarrow 1 \rightarrow 2 \rightarrow 3$

LIST HALF SWAP (*list*)