

Strukture podataka

April 2019

1. (25 poena) Polje, Niz, Magacin, Red, Lančane liste
 - a. (12 poena) Ako su dostupne osnovne operacije za obradu reči (izdvajanje podniza, indeksiranje, konkatencija i nalaženje dužina niza), napisati pseudokod operacije koja predstavlja modifikaciju kompozitne operacije za umetanje u tekst T, tako što se umesto na poziciju k, novi niz S dodaje na poziciju od koje počinje zadati uzorak P.
 - a. (13 poena) Napisati funkciju **void UmetniListu(DLList uredjena, DLList nova)** koja elemente neuređene dvostruko ulančane liste *nova* umeće u dvostruko ulančanu listu *uredjena* tako da ova lista ostane uređena u rastućem redosledu. Prilikom dodavanja svakog sledećeg elementa iz liste *nova* traženje mesta započeti od pozicije na kojoj se nalazi prethodno umetnuti element u listi *uredjena*.
2. (25 poena) Heš tablice
 - b. (12 poena) Opisati proces traženja elementa iz Heš tablice ako se kao primarna transformacija koristi heš funkcija $h(k)$, a kao sekundarna modifikovano linearno traženje mesta za sinonime. Na koji način se tretira specijalna vrednost DELETED kod traženja?
 - c. (13 poena) Skeniranjem realnog prostora dobijena je 3D matrica dimenzija $M \times N \times P$. Najveći broj podataka iz matrice je jednak nuli, pa se za njeno smeštanje koristi rasuta tablica sa otvorenim adresiranjem. Napisati: konstruktor, primernu transformaciju, sekundarnu transformaciju i funkciju za dodavanje vrednosti u tablicu **void insert(int x, int y, int z, double val)**.
3. (25 poena) Stabla
 - a. (13 poena) Napisati pseudokod operacije za kreiranje stabla binarnog traženja S na osnovu elemenata niza A. Koristiti sekvencijalnu reprezentaciju stabla. Prikazati jedno stablo binarnog traženja sa najmanje 5 čvorova i prikazati njegovu sekvencijalnu memorijsku reprezentaciju.
 - b. (12 poena) Napisati funkciju **int LeavesCount()**, koja vraća broj listova statički implementiranog binarnog stabla. Smatrati da su ključevi stabla prirodni brojevi, a da -1 označava nepopunjeno mesto.
4. (25 poena) Grafovi
 - a. (12 poena) Za lančanu reprezentaciju orijentisanog grafa, napisati pseudokod operacije koja određuje ulazni stepen $indeg(u)$ datog čvora u . Pseudokod glavnog programa treba da prikaže samo one čvorove koji imaju $indeg(u)=0$.
 - b. (13 poena) Napisati funkciju **bool CanReach(Node* a, Node* b, int hops)** koja proverava da li je moguće stići iz čvora a do čvora b dinamički implementiranog usmerenog grafa. **Napomena:** Rešenja koja vrše izračunavanje dužine svih puteva u grafu će biti ocenjivana sa 50% poena.