L04 - rešenja zadataka

Zadatak 1. Stabilnost sortiranja

Napomena: niz A je ulazni niz; niz B je izlazni, sortiran niz; niz C je pomoćni niz određen algoritmom counting sort-a; indeksi svih nizova idu od 0 do dužine niza umanjenu za 1; svi elementi niza imaju vrednost u intervalu [0,k)

Neka je dat niz A od n elemenata, koji sadrži element $x \in [0, k)$, koji se unutar niza A ponavlja ukupno p puta. Polazeći od početka niza A ka njegovom kraju, pridružimo svakom od elemenata vrednosti x satelitski podatak koji će predstavljati koliko se elemenata vrednosti x pojavilo pre njega samog[1] (važi $ix_0 < ix_1 < ix_2 < ... < ix_{p-1}$). Ovim će vrednost satelitskog podatka elemenata vrednosti x monotono rasti sa porastom indeksa elemenata vrednosti x (isto važi i u obrnutom smeru) **[1]**.

...	x	x	...	x	...	x	...
	0	1		2		$p - 1$	
	ix_0	ix_1		ix_2		ix_{p-1}	

Niz A

Niz C će nakon brojanja pojavljanja elemenata i brojanja elemenata manjih od ili jednakog za svaki od elemenata izgledati na sledeći način, uz pretpostavku da postoji m elemenata manjih od x:

	...	$m + p$	...	
0		x		$k - 1$

Niz C

Po algoritmu, elementi se pri smeštanju u niz B čitaju s kraja niza A, odnosno prvo se čita element na indeksu $n - 1$, sve do elementa sa indeksom 0. Odavde se može zaključiti da će se od svih elemenata vrednosti x, prvo pročitati onaj kom je pridružen satelitski podatak $p - 1$, pošto tekući indeks petlje pri čitanju podataka iz niza A monotono opada sa korakom 1 tako će i opadati vrednost satelitskog podatka **[1]**, i tako redom sve dok se ne pročita i element kom je pridružen satelitski podatak vrednosti 0. *(Napomena: neće doći do "preskakanja" nekog od elemenata niza A pri čitanju elemenata baš zato što indeks čitanja monotono opada sa korakom 1 - ovim se osigurava da će i indeksi elemenata vrednosti x monotono opadati prolaskom kroz petlju, tj. od svih elemenata vresnoti x, prvi od njih koji bude bio pročitan će biti onaj sa indeksom ix_{p-1}, nakon njega sa indeksom ix_{p-2}, itd. i na kraju onaj sa indeksom ix_0.)* Prateći algoritam, element $(x, p - 1)$ će u niz B biti stavljen na mesto $m + p - 1$, C[x] će biti umanjeno za 1. U nekoj od narednih iteracija biće pročitan element $(x, p - 2)$ i biće stavljen na mesto $m + p - 2$ u nizu B, C[x] će biti umanjen za 1, itd. sve dok se ne pročita element $(x, 0)$ koji će biti smešten na mesto m.

...	x	x	x	x	x	...
	0	1	...	$p - 2$	$p - 1$	
	m	$m + 1$	...	$m + p - 2$	$m + p - 1$	

Niz B

1 ovakve vrednosti satelitskih podataka su korišćene kako bi se elementi istih vrednosti razlikovali i kako bi se označio njihov redosled pojavljivalja u nizu, radi lakšeg razumevanja dokaza

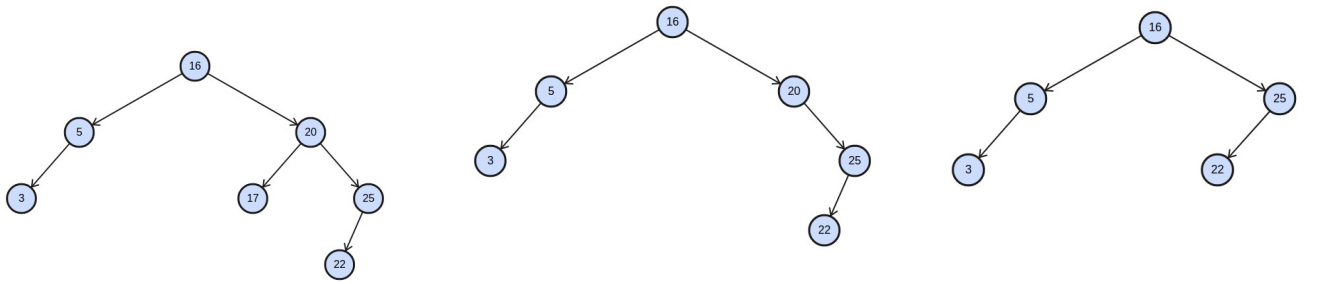
Counting sort bi ostao korektan kao algoritam sortiranja ukoliko bi se obrnuo redosled čitanja elemenata iz niza A, međutim, izgubio bi osobinu stabilnosti, jer bi elementi bili stavljeni u niz B obrnutom redosledu od redosleda pojavljivanja u nizu A (odnosno da bi uz ovu promenu ostao stabilan, bilo bi potrebno modifikovati ostatak algoritma/pseudokoda).

Zadatak 2. Brisanje elementa iz BST

Operacija brisanja elementa iz BST ***nije*** komutativna operacija. Kontraprimer:

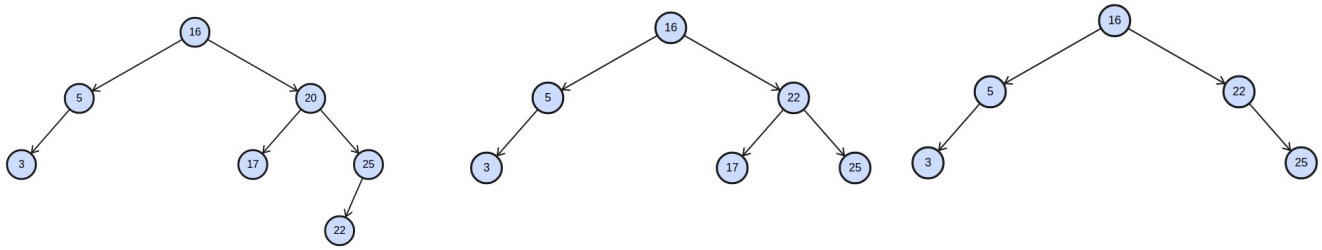
- *Iz sledećeg BST prvo ćemo obrisati element 17, pa element 20:*

1. Originalno stanje *2. Brisanje elementa 17* *3. Brisanje elementa 20*



- *Sada ćemo iz istog takvog BST obrisati element 20, pa element 17:*

1. Originalno stanje *2. Brisanje elementa 20* *3. Brisanje elementa 17*



U ovom primeru, dobijen je ovakav rezultat zato što je u prvom slučaju element 17 je bio list (samo se ukloni iz stabla) i jedini element u levom podstablu elementa 20, tako da je element 20 sada ostao bez svog levog deteta i celog levog podstabla, što utiče na to da kada uklonimo element 20, on će biti zamenjen svojim preostalim desnim detetom, elementom 22. U drugom slučaju, prvo smo obrisali element 20, a pošto on ima oba svoja deteta, on će biti zamenjen svojim naslednikom, elementom 25. Dalje, briše se element 17 koji se samo uklanja stabla jer je on list. Razlog različitih rezultata ova dva slučaja jeste to što se element koji se briše iz stabla ne zamenjuje sa istim elementom u oba slučaja usled drugačije strukture stabla pre njegovog brisanja.