

**CS103 - DOMAĆI ZADATAK 08.**

Prilikom slanja domaćeg zadatka svom asistentu neophodno je da ispunite sledeće:

* Subject mail-a mora biti CS103-DZbr. Za ovaj domaći zadatak - CS103-DZ08
* Sve fajlove, koji su deo rešenja zadataka, arhivirati (zip, rar, …)
* Poželjno je uraditi i printscreen koda pre pokretanja programa
* U prilogu maila treba da se nalazi arhiva projekta koji se ocenjuje, imenovana na sledeći način: CS103-DZbr-BrojIndeksa-ImePrezime*. Na primer, CS103-DZ08-9999-VeljkoGrkovic.zip*
* Telo mail-a treba da ima pozdravnu poruku

**Molimo sve studente da se pridržavaju navedenog, inače zadaci neće biti pregledani i ocenjeni.**

Studenti iz Beograda i online studenti zadatke na ocenjivanje šalju mail-om na adresu [lazar.mrkela@metropolitan.ac.rs](mailto:lazar.mrkela@metropolitan.ac.rs)

Studenti iz Niša zadatke na ocenjivanje šalju mail-om na adresu  
[nikola.dimitrijevic@metropolitan.ac.rs](mailto:nikola.dimitrijevic@metropolitan.ac.rs)

**Rok za slanje domaćih zadataka:**

* **Tradicionalni studenti: sedam dana od dana održavanja vežbe na koju se odnosi domaći zadatak. Nakon toga, broj poena se umanjuje za 50%. Krajnji rok za slanje rešenja je deset dana pre termina ispita u kome želite da polažete ispit.**
* **Studenti na online nastavi, kao i studenti kojima je predmet prenet u narednu godinu ili je diferencijalni: najkasnije deset dana pre termina ispita u kome želite da polažete ispit.**

*Svaki student radi jedan zadatka sa spiska*. Ostali zadaci mogu da posluže za vežbanje i pripremu ispita, ali ih ne šaljete na pregled asistentima.

Redni broj zadatak dobijate tako što vaš broj indeksa podelite sa 13, a dobijeni ostatak pri deljenju uvećate za 1. Na primer:   
Broj indeksa 2378  
2378 % 13 + 1 = 7 – student radi 7. zadatak.

1. Implementirati inorder metod u binarnom stablu pretrage (BST) korišćenjem steka umesto rekurzije.
2. Implementirati preorder metod u binarnom stablu pretrage (BST) korišćenjem steka umesto rekurzije.
3. Implementirati postorder metod u binarnom stablu pretrage (BST) korišćenjem steka umesto rekurzije.
4. Dodati metodu u binarnom stablu pretrage (BST) koja računa broj listova stabla   
   public int getNumberOfLeaves().
5. Dodati metodu u binarnom stablu pretrage (BST) koja računa broj čvorava koji nisu listovi stabla public int getNumberofNonLeaves().
6. Dodati metodu u binarnom stablu pretrage (BST) koja vrši levu rotaciju (LL rotation) u cilju balansiranja stabla.
7. Dodati metodu u binarnom stablu pretrage (BST) koja vrši desnu rotaciju (RR rotation) u cilju balansiranja stabla.
8. Dodati metodu u binarnom stablu pretrage (BST) koja vrši levu-desnu rotaciju (LR rotation) u cilju balansiranja stabla.
9. Dodati metodu u binarnom stablu pretrage (BST) koja vrši desnu-levu rotaciju (LR rotation) u cilju balansiranja stabla.
10. Napisati program u Java-i koji proverava da li je stablo AVL ili ne.
11. Napisati metodu za binarno stablo pretrage (BST) koja uzima dva ključa, low i high, I štampa sve elemente stabla koji su u opsegu između low i high. U nekom od komentara u kodu navesti vreme izvršavanja vašeg algoritma (big O notacija).
12. Implementirati binarno stablo pretrage (BST) koje dozvoljava duplikate. U tu svrhu kreirati klasu za čvor stabla koja će osim referenci Levi i Desni, imati i referencu na povezanu listu u kojoj će se čuvati duplikati.
13. Dodati metodu u binarnom stablu pretrage (BST) koja vrši brisanje elementa iz binarnog stabla pretrage. Napisati glavni program u cilju testiranja, kao i klasu za BST.