

Ime i prezime: \_\_\_\_\_

\* Minimum za prolaz je u potpunosti riješiti DVIJE točke iz praktičnog dijela i ZADATAK 1-a).

### Zadatak 1.

Zadan je niz brojeva: 47, 16, 62, 42, 48, 44, 17, 64, 93, 22, 10, 5.

**a \*) Upisati brojeve redom u binarno uređeno stablo (nacrtati sliku na papir). Ako stablo ne bude ispravno, točke c i d će također biti 0.**

b) Ispisati čvorove po PREORDER poretku. Rješenje:

\_\_\_\_\_

c) Nacrtati kako izgleda stablo kad se izbriše čvor: 16.

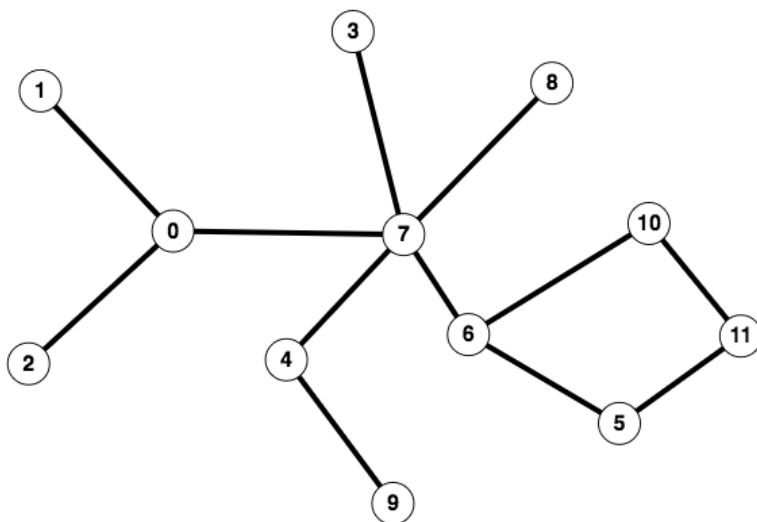
Rješenje a)

Rješenje c)

### Zadatak 2.

Zadan je graf kao na slici. Ispisati popis čvorova pomoću pretrage po dubini počevši od čvora 2.

**Ispisati svaki posjećen čvor samo jednom** (kad se prvi put posjeti). Prednost pri odabiru puta imaju čvorovi s manjim brojem.



## Praktični dio:

**Napomene: Točke koje ne ovise jedna o drugoj se ne moraju rješavati po redu.**

0. Program ne smije prestajati raditi («pucati» zbog iznimke ili sintaksne pogreške). Programi koji se ne mogu pokrenuti neće biti pregledani. Svaka točka mora u potpunosti raditi da bi bila priznata (osim za kolokvij). Sve što se preda je konačna verzija, te nema ispravljanja nakon što završi ispit/kolokvij. Ako rješavate dio sa stablima, **stablo se mora ispravno nacrtati**.

Datoteka „**cvorovi.txt**“ sadrži popis čvorova, a „**veze.txt**“ sadrže informacije o vezama – za graf.

1. Iz datoteke **cvorovi.txt** treba učitati podatke o učenicima u uređeno binarno stablo. Podaci o svakome sastoje se od: *imena* i jedinstvenog *ID broja*. Uređaj se temelji na tom broju. Nisu dozvoljeni duplikati pa će se spremati **zadnji** koji se pojavljuje u datoteci, a ispisat će se imena duplikata koji nisu spremljeni u stablo. Napisati metodu **Pre()** unutar klase **BinarnoStablo** koja ispisuje stablo (broj i naziv) po preorder poretku.
  - a. Obavezno riješiti zadatak 1. a).
2. Napisati metodu **Put()** unutar klase **BinarnoStablo** koja za traženi čvor vraća popis čvorova koje treba proći od korijena do traženog čvora. Korisnik upisuje broj čvora kojeg traži u **textBox**, ako je ispravno uneseno ispisuje put kojim se dolazi do njega (dovoljno je samo brojeve) ili poruku ako ga nema.
3. Učitati sve čvorove u **listBox** na način koji želite, a koristit će se za odabir čvorova za brisanje. Napisati metodu **BrisiMe()** unutar klase **BinarnoStablo** koja pokreće brisanje odabranog čvora pod uvjetima: a) čvor će se pobrisati ako je list, b) ako nije list ispisati podatke o njegovom roditelju (ime i broj). Nakon brisanja **obavezno osvježiti samo sliku**, u **listBoxu** će ostati pobrisani čvorovi.
4. Učenici su se tijekom sprijateljili, a njihova prijateljstva su zapisana u datoteci **veze.txt**. Napisati klasu/strukturu **Graf** koja omogućuje unos čvorova i veza u neusmjeren graf. Učitati sve podatke u strukturu grafa i ispisati na **richTextBox**: popis čvorova i veza među njima. Ispisati graf tako da se ispisuje čvorove te za svaki čvor popis prijatelja ispod njega (pazite da bude pregledno).
5. Ispisati koji učenik ima najviše prijatelja. Počevši od **drugog** upisanog čvora ispisati sve čvorove na **richTextBox** pomoću algoritma pretrage po širini (naziv i broj).