

U direktorijumu **Rad** koji se nalazi na desktopu kreirati direktorijum **Ime\_prezime\_indeks** i u okviru njega kreirati c fajlove koji sadrže programe koji rešavaju sledeće probleme, pri čemu ime fajla MORA DA ODGOVARA „imenu“ zadatka.

### **Prvi:**

- Potrebno je demonstrirati smeštanje podataka u okviru fajl sistema. Jedan fajl se, usled nedostatka mesta na sistemu, čuva po delovima, gde se za svaki deo fajl sistema pamte sledeći podaci:
  - ✓ Indikator o zauzeću (1,0 – celi brojevi)
  - ✓ Naziv fajla (više reči)
  - ✓ ID fajla
  - ✓ Veličina (u KB, realan broj)
- Inicijalno stanje fajl sistema se pravi na taj način što se unose podaci o delovima. Ne zna se koliko ima delova fajl sistema, već se podaci učitavaju dok se za indikator ne unese ceo broj različit od 0 i 1. Za svaki deo fajl sistema se najpre unosi indikator o zauzeću (1-zauzet, 0-slobodan). Ukoliko je deo zauzet dodatno se učitava naziv fajla i ID fajla, u svakom slučaju se učitava veličina tog dela fajl sistema u KB. Podaci se čuvaju u okviru jednostruko povezane liste u redosledu kako se unose.
- Ispisati kreirano stanje fajl sistema.
- Zatim se radi delimična fragmentacija fajl sistema na taj način što se svi delovi jednog istog fajla smeštaju na uzastopne lokacije i to tamo gde se on prvi put javlja.
  - Ispisati stanje fajl sistema nakon fragmentacije.
  - Zatim sve uzastopne elemente liste koji predstavljaju delove istog fajla zameniti samo jednim elementom, čija je veličina ukupna veličina celog fajla.
  - Ispisati stanje fajl sistema nakon fragmentacija.
- Nakon fragmentacije, vrši se defragmentacija. Na ulazu se učitava broj K a zatim podaci o K fajlova koje je potrebno smestiti u okviru fajl sistema (može da se desi da ne se ne smeste svi fajlovi jer nema mesta). Za svaki fajl se učitava naziv fajla, ID fajla i ukupna veličina, koja se deli na slobodna mesta u okviru fajl sistema i vrši se zauzimanje prostora redom.
- Ispisati kreirano stanje fajl sistema nakon defragmentacije.

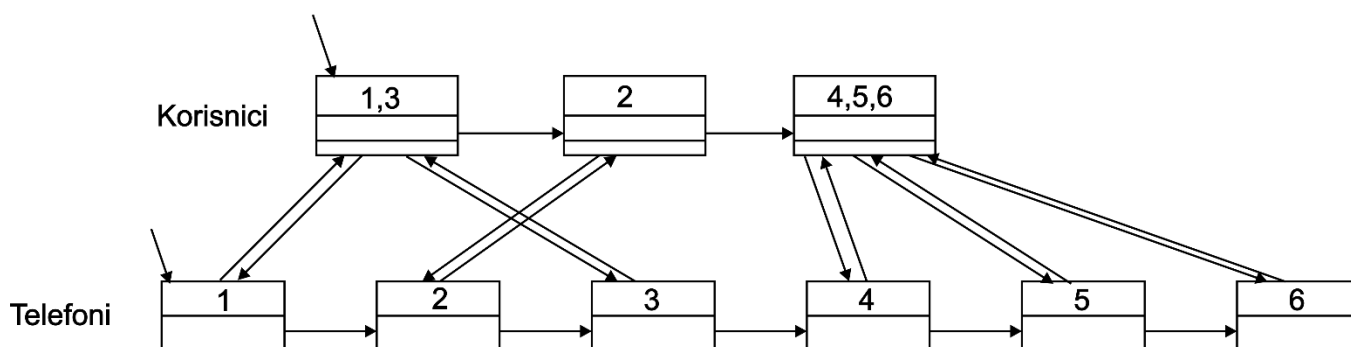
NAPOMENA: Nije dozvoljeno proširivati strukturu, niti koristiti nizove.

### Drugi1:

Organizuje se baza **mobilne telefonije IMI**. Za svakog korisnika se pamti **Ime** (jedna reč), **broj mobilnih brojeva** (ceo broj), **niz ID-a** mobilnih brojeva (niz celih brojeva) i niz pokazivača ka mobilnim brojevima (jedan korisnik ne može imati više od tri broja). Za svaki mobilni broj se pamti **ID\_broja** (ceo broj), sam **broj** (niz karaktera) i pokazivač ka korisniku broja (jedan broj može imati samo jednog korisnika). U programu se najpre unosi broj **n** koji predstavlja broj korisnika a zatim se za svakog korisnika unosi ime, koliko ima mobilnih brojeva i ID-jevi tih mobilnih brojeva. Podaci o korisnicima se čuvaju u jednostruko povezanoj listi. Zatim se unosi ceo broj **m** koji predstavlja broj mobilnih brojeva koji se koriste i za svaki broj se unosi samo njegova vrednost (niz karaktera), dok se ID dodeljuje po redosledu unošenja. Podaci o mobilnim brojevima se takođe čuvaju u jednostruko povezanoj listi. Obe liste se formiraju dodavanjem novog elementa na kraj.

Zatim se vrši međusobno povezivanje korisnika sa mobilnim brojevima, na osnovu ID-a koji korisnici imaju. Korisnik ne može imati nepostojeći ID i o tome nije potrebno voditi računa. Nakon povezivanja odštampati podatke o svim korisnicima, tako što se podaci o jednom korisniku štampaju u jednom redu zajedno sa podacima o njegovim mobilnim brojevima. Zatim se unosi **K** mobilnih brojeva (niz karaktera) koji predstavljaju mobilne brojeve koji se isključuju zbog neplaćenog računa. U tom slučaju brojevi se brišu iz liste mobilnih brojeva, kao i iz niza korisnika broja. Ukoliko je korisniku uklonjen njegov jedini broj i on se briše iz liste korisnika. Nakon ažuriranja ponovo odštampati podatke o svim korisnicima, tako što se podaci o jednom korisniku štampaju u jednom redu zajedno sa podacima o njegovim mobilnim brojevima.

NAPOMENA. Pri unosu poštovati navedeni redosled podataka.

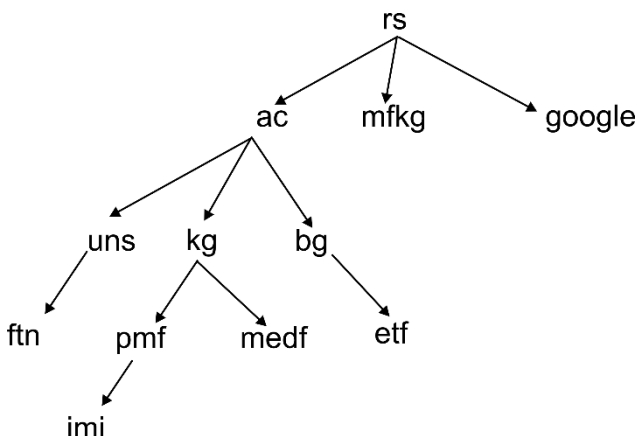


### Drugi2:

Na ulazu se zadaje ceo broj **N** a zatim **N** reči koje predstavljaju web stranice koje su sigurno pod domenom **rs**. Naziv svake unete stranice se parsira i formira stablo. Parsiranje se vrši odvajanjem delova razdvojenih tačkom, na primer **imi.pmf.kg.ac.rs** se razdvaja na pet delova, **imi**, **pmf**, **kg**, **ac** i **rs**, gde je svaki deo informacija u okviru čvora. Formiranje stabla je prikazano na slici. Jedan čvor ne može imati više od 10 naslednika. Odštampati formirano stablo s leva na desno. Napisati funkciju koja za dati naziv stranice proverava da li se nalazi u datom stablu.

Ovaj primer je imao sledeći ulaz:

7  
ftn.uns.ac.rs  
imi.pmf.kg.ac.rs  
google.rs  
etf.bg.ac.rs  
mfkg.rs  
medf.kg.ac.rs  
pmf.kg.ac.rs



### **Treći:**

Korišćenjem dinamičkih struktura podataka stabla (Crveno-Crna maksimum poena je 20, ukoliko se zadatak reši korišćenjem AVL stabla maksimum je 17 poena) sastaviti program koji treba da organizuje pravljenje ekipa na osnovu brojeva koje deca pojedinačno prijavljuju. Prvo se unosi broj dece **n**, a zatim se unose podci o deci. Za svako dete se pamti ime (jedna reč), prezime (jedna reč), broj izabranih brojeva (ceo broj) i niz izabranih celih brojeva (brojevi koje dete bira se ne ponavljaju). U svrhu lakše organizacije koristi se struktura podataka pogodna za bržu pretragu, tj. u ovom slučaju binarno pretraživačko stablo.

Stablo se uređuje prema imenu, a ako je ime isto, prema prezimenu.

- Nakon završenog rasporeda štampati podatke o deci od A-Z.
- Napisati rekurzivnu funkciju koja **za dati koren stabla** pronalazi dete čija je srednja vrednost izabranih brojeva najbliža broju M.
- U glavnom delu programa štampati podatke o prethodno pronađenom detetu.
- Sada je potrebno organizovati ekipe i svu decu koja imaju zajedničke prijavljene brojeve staviti u jednu ekipu čije je ime sastavljeno od početnih slova imena sve dece te ekipe a prezime sastavljeno od početnih slova prezimena sve dece te ekipe, dok je niz presek tih prijavljenih brojeva. Ukoliko neko dete nema zajedničkih prijavljenih brojeva ni sa kim, on ostaje kao samostalan takmičar. (Brisanje bez kreiranja novog stabla donosi bonus od 3 poena.)
- Štampati podatke o ekipama od A-Z.

NAPOMENA: Obavezno koristiti funkciju za štampanje po nivoima. Takođe, koristiti i štampu datu u zadatku.