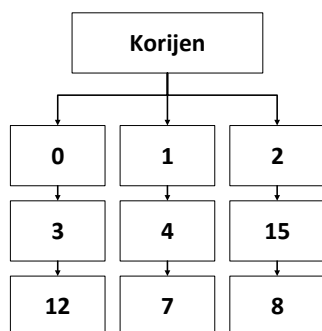


## PROGRAMIRANJE II (4.9.2015.)

- ❶ (50 bodova) Realizovati strukturu „modularnog skupa“ koje radi sa prirodnim brojevima. Struktura „modularni skup“ predstavlja skup podataka koji se sastoji iz tri liste koje se nadovezuju na jedan centralni čvor. Tri liste imaju osobinu da  $i$ -talista može da sadrži samo elemente čiji je ostatak pri dijeljenju sa 3 jednaki, pa se tako dodavanje u strukturu realizuje kao dodavanje u odgovarajuću listu od tri moguće. Korijenski čvor predstavlja slog (eng. *struct*) sa tri pokazivača na početak svake od lista i predstavlja slog nad kojim se pozivaju funkcije za dodavanje, izbacivanje, sortiranje i ispis elemenata. Slika 1 ilustruje izgled ovakve strukture, a u Tabeli 1 su date funkcije koje je neophodno implementirati. Iz zaglavlja funkcija se mogu zaključiti nazivi slogova koje treba koristiti pri implementaciji. Liste treba realizovati kao jednostruko uvezane liste. U glavnom programu treba ilustrovati rad svake od navedenih funkcija koje rade sa korjenskim čvorom modularnog skupa.

Tabela 1

Zaglavlje	Specifikacija
<code>dodaj(KORJENSKI_CVOR*, int)</code>	Dodaje element u odgovarajuću listu pozivom funkcije za dodavanje u listu.
<code>dodaj_u_listu(CVOR_LISTE*, int)</code>	Dodaje element u listu definisanu početnim čvorom liste, pod uslovom da taj element prethodno ne postoji u listi.
<code>izbaci(KORJENSKI_CVOR*, int)</code>	Izbacuje element iz odgovarajuće liste pozivom funkcije za izbacivanje iz liste, ako on postoji.
<code>izbaci_iz_liste(CVOR_LISTE*, int)</code>	Izbacuje iz liste definisane početnim čvorom element sa datom vrijednošću.
<code>sortiraj(KORJENSKI_CVOR*)</code>	Sortira sve tri liste na koje pokazuje korjenski čvor, pozivom funkcija za sortiranje liste.
<code>sortiraj_listu(CVOR_LISTE*)</code>	Sortira listu određenu početnim čvorom.
<code>ispisi(KORJENSKI_CVOR*)</code>	Ispisuje sadržaje lista redom po njihovom modularnom indeksu pozivom funkcija za ispis pojedinačne liste.
<code>ispisi_listu(CVOR_LISTE*)</code>	Ispisuje sadržaj liste određene početnim čvorom.



Slika 1

- ❷ (50 bodova) Napisati program koji iz ulazne tekstualne datoteke čita dimenzije matrice susjednosti, a potom matricu susjednosti težinskog, usmjerenog grafa. Težine su definisane kao podaci sa pokretnim zarezom, pri čemu se garantuje da su težine pozitivne, te da je graf potpuno povezan. Naziv ulazne datoteke kao i indeksi dva čvora unose se kao argumenti komandne linije. Na standardni izlaz ispisati funkcije vrijednosti čvorova na najkraćoj putanji između čvorova čiji su indeksi uneseni kao argumenti komandne linije. Funkcije koje treba izračunati za svaki čvor su:  $\sin(x)$ ,  $\cos(x)$ ,  $x^2$ ,  $\sqrt{x}$ . Ispis uraditi tako da se funkcija za pronalaženje najkraćeg puta između data dva čvora poziva četiri puta, svaki put sa različitom funkcijom za ispis u argumentu (funkcija za pronalaženje najkraćeg puta prihvata funkciju za ispis kao argument).

Napomena: za svaku od datih matematičkih funkcija, potrebno je napisati odvojenu funkciju sa prefiksom `t_` (`t_sin` za `sin`, `t_cos` za `cos`, `t_pow2` za `pow` i `t_sqrt` za `sqrt`) kako bi funkcije koje se prosljeđuju kao argumenti imale isto zaglavlje. Za njihovo pojedinačno realizovanje dozvoljeno je korištenje biblioteke *math.h*.