1. Definisati klasu **Racun** sa funkcijama za zbir, razliku, proizvod i količnik dva broja. U glavnom programu kreirati  jedan objekat klase Racun i ponuditi korisniku mogućnost izbora željene operacije (switch) kako bi se ispisao odgovarajući rezultat. Korisniku se omogućava da po želji ponavlja unos (petlja) sve dok ne unese x, kada se program završava.

2. Frižideri se opisuju sa dva privatna atributa, zapreminom raspoloživog prostora i potrošnjom električne energije na sat. U glavnom programu uneti niz objekata klase frizider, sortirati ih u neopadajućem redosledu po efikasnosti, i odštampati sortirani niz. Na kraju programa obavezno obrisati sve sto je kreirano u dinamičkoj zoni memorije.

3. U header fajlu kvadar.h deklarisati klasu **Kvadar** sa celobrojnim privatnim članovima a, b, c i funkcijama za zapreminu (V=abc) i površinu (O = 2(ab+ac+cb)). Definisati tri konstruktora (bez parametara, sa jednim parametrom i sa tri parametra) te njima, u glavnom programu, inicijalizovati objekte klase **Kva1**, **Kva2**, **Kva3**.  Vrednost parametara konstruktora unosi korisnik u glavnom programu. Za svaku od instanci ispisati vrednost zapremine i površine.

4. Definisati klasu koja modeluje Red celih brojeva i kao privatni atribut ima kapacitet niza. Red je FIFO struktura podataka i ima sledeće funkcije:

- Enqueue - dodaje novi element na početak niza (vraća 1 ako je elemenat uspešno dodat a 0 ako je red pun)

- Dequeue - vraća zadnji elemenat izbacuje ga iz niza (ako je niz prazan vraća null)

- bool Redim(int novoN) - menja maksimalni broj elemenata na novoN i ponovo zauzima memoriju. Ako je trenutni broj elemenata u nizu veći od novoN, funkcija nema efekta.

U glavnom programu instancirati nekoliko objekata klase Red i testirati sve kreirane funkcije.

5. Definisati klasu koja modeluje niz celobrojnih elemenata i ima sledeće funkcionalnosti:

- Kreiranje niza koji ima n elemenata

- Kreiranje niza koji ima n2 elemenata

- Dodavanje novog elementa u niz na prvo slobodno mesto

- Izbacivanja elementa iz niza sa navedenog mesta

- Funkciju koja kao ulaz ima niz indeksa, a kao izlaz vraća niz istih indeksa uredjenih tako da prate povezane vrednosti u nizu uredjene u nerastuci poredak

Korišcenjem ove klase rešiti problem „25 konja“:

Let’s say that you have 25 horses, and you want to pick the fastest 3 horses out of those 25. In each race, only 5 horses can run at the same time because there are only 5 tracks. What is the minimum number of races required to find the 3 fastest horses without using a stopwatch?