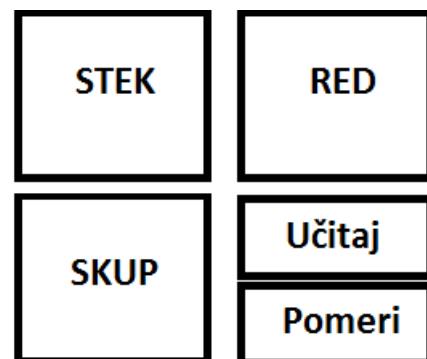


2. **(JavaThreads 50%)** Server može da opsluži najviše jednog klijenta istovremeno (iako je u praksi taj broj mnogo veći, obično i po nekoliko hiljada). Postoji ukupno 100 klijenata, i svaki od njih pokušava da se obrati serveru na svakih 100 milisekundi. Svaki klijent se opisuje svojim kapacitetom preuzimanja (download bandwidth) u Mb/s. Ti kapaciteti su između 1 i 10Mb/s (celobrojni nasumični). Server ima kapacitet davanja dovoljno veliki (10Mb/s+, tako da je dovoljno da svaki klijent preuzima punom brzinom). Kada se klijent „zakači“ na server, on radi između 10 i 100 milisekundi (nasumično vreme, koristiti Thread.sleep u okviru kritične sekcije za ovo), i za to vreme preuzme odgovarajući broj podataka (npr. ako radi 50 ms i ima download od 4Mb/s, on će preuzeti $4\text{Mb/s} \times 50\text{ms} = 4\text{Mb/s} \times 1\text{s}/20 = 4/20 \text{ Mb} = 0.2\text{Mb}$). Svaki klijent pamti koliko je dosad preuzeo, a server pamti koliko je dao. Suma svih preuzetih je jednaka u svakom momentu količini koju je server dao (invarijanta). Napraviti monitor klasu koja svake sekunde ispisuje ukupni prenos od strane servera i ukupni prenos ka svim klijentima (ove dve stvari, dakle, moraju biti iste u svakom momentu).

1. **(JavaFX 50%)** Prozor inicijalno ima tri prazne grafičke komponente liste (u gornjem levom i desnom i donjem levom uglu). U donjem desnom uglu su dva dugmeta, „Učitaj“ i „Pomeri“. Dugme „Učitaj“ učitava sadržaj datoteke „imena.txt“ u kojoj se nalaze sledećih 12 imena: Ana, Mitar, Petar, Mira, Marija, Marko, Stefan, Nevena i Marko (opet), Stefan (opet), Andrija i Jovana. Učitavanje se vrši tako da se redom popuni sa po 4 elementa svaka od lista. Svaka lista se ponaša na drugačiji način, naime, prva se ponaša kao stek, druga kao red, a treća kao skup. Podsetnik: kod steka se element stavlja na početak i skida sa početka, kod reda se stavlja na početak, a skida sa kraja, a kod skupa neka bude redosled kao kod reda, jedino što ne postoji mogućnost postojanja duplikata. Pritiskom na dugme „Pomeri“ vrši se jedan ciklus pomeranja imena: najpre se iz steka skida ime i ubacuje u red, iz reda se skida i ubacuje u skup, a iz skupa se skida i ubacuje u stek.



2. **(JavaThreads 50%)** Server može da opsluži najviše jednog klijenta istovremeno (iako je u praksi taj broj mnogo veći, obično i po nekoliko hiljada). Postoji ukupno 100 klijenata, i svaki od njih pokušava da se obrati serveru na svakih 100 milisekundi. Svaki klijent se opisuje svojim kapacitetom preuzimanja (download bandwidth) u Mb/s. Ti kapaciteti su između 1 i 10Mb/s (celobrojni nasumični). Server ima kapacitet davanja dovoljno veliki (10Mb/s+, tako da je dovoljno da svaki klijent preuzima punom brzinom). Kada se klijent „zakači“ na server, on radi između 10 i 100 milisekundi (nasumično vreme, koristiti Thread.sleep u okviru kritične sekcije za ovo), i za to vreme preuzme odgovarajući broj podataka (npr. ako radi 50 ms i ima download od 4Mbps, on će preuzeti $4\text{Mb/s} \times 50\text{ms} = 4\text{Mb/s} \times 1\text{s}/20 = 4/20 \text{ Mb} = 0.2\text{Mb}$). Svaki klijent pamti koliko je dosad preuzeo, a server pamti koliko je dao. Suma svih preuzetih je jednaka u svakom momentu količini koju je server dao (invarijanta). Napraviti monitor klasu koja svake sekunde ispisuje ukupni prenos od strane servera i ukupni prenos ka svim klijentima (ove dve stvari, dakle, moraju biti iste u svakom momentu).

1. **(JavaFX 50%)** Prozor inicijalno ima tri prazne grafičke komponente liste (u gornjem levom i desnom i donjem levom uglu). U donjem desnom uglu su dva dugmeta, „Učitaj“ i „Pomeri“. Dugme „Učitaj“ učitava sadržaj datoteke „imena.txt“ u kojoj se nalaze sledećih 12 imena: Ana, Mitar, Petar, Mira, Marija, Marko, Stefan, Nevena i Marko (opet), Stefan (opet), Andrija i Jovana. Učitavanje se vrši tako da se redom popuni sa po 4 elementa svaka od lista. Svaka lista se ponaša na drugačiji način, naime, prva se ponaša kao stek, druga kao red, a treća kao skup. Podsetnik: kod steka se element stavlja na početak i skida sa početka, kod reda se stavlja na početak, a skida sa kraja, a kod skupa neka bude redosled kao kod reda, jedino što ne postoji mogućnost postojanja duplikata. Pritiskom na dugme „Pomeri“ vrši se jedan ciklus pomeranja imena: najpre se iz steka skida ime i ubacuje u red, iz reda se skida i ubacuje u skup, a iz skupa se skida i ubacuje u stek.

