

## **Zadatak: Mobilni Operater**

Napisati Java konzolnu aplikaciju kojom se opisuje rad mobilnih operatera u Srbiji. Svaki mobilni operater može imati korisnike kao fizička lica, ili kao pravna lica (firme), pri čemu jedno pravno lice ima određeni broj korisnika. Korisnici mogu koristiti tri vrste usluga mobilnih operatera: razgovor, slanje poruka ili Internet. Svaki operater definiše cenu minuta razgovora, cenu jedne poruke i cenu jednog MB Internet saobraćaja. Korisnici mogu biti prepaid ili postpaid korisnici. Prepaid korisnike karakteriše unapred određen iznos kredita koji se može trošiti, koristeći usluge operatera. Postpaid korisnike karakteriše tarifni paket koji ima svoju fiksnu cenu i podrazumeva određeni broj minuta, poruka i Internet saobraćaja koji je uračunat u tu fiksnu cenu. Dodatno, ukoliko korisnik koristeći usluge operatera prekorači minute, poruke ili Internet saobraćaj u okviru paketa, obračunava mu se dodatna naplata (prema cenama operatera kojem pripada) i koja se čuva u posebnom atributu stanje. U jednom trenutku (na kraju meseca), operater obračunava račun za naplatu postpaid korisnika, sabirajući fiksnu cenu tarifnog paketa sa iznosom za koji je prekoračen sam tarifni paket. Pravna lica mogu imati samo postpaid korisnike i to samo one korisnike sa biznis tarifnim paketima (boolean atribut `biznis` u klasi `TarifniPaket`). Za pravna lica se takođe može generisati račun koji čini zbir svih računa postpaid biznis korisnika koji su deo tog pravnog lica. Dodatno, svaki postpaid korisnik može kupiti određeni tarifni dodatak koji može biti SMS ili Internet dodatak i koji ima svoju fiksnu cenu, koja se takođe uračunava u konačni račun. Takav dodatak omogućava besplatno slanje poruka, odnosno surfovanje Internetom. Svaki razgovor, poslata poruka ili Internet sesija koju pojedinačni korisnik napravi se beleži.

Potrebno je predstaviti model identifikovanih klasa i interfejsa i implementirati ih zajedno sa pratećim `get` i `set` metodama. Dodatno je potrebno implementirati i sledeće metode:

### **Mobilni Operater:**

**generisiRacune()** – generiše sve račune za svako postpaid fizičko lice, kao i za svako pravno lice pozivom metoda `generisiRacun()` nad njima i sve račune ispisuje na standardni izlaz (konzolu)

**prikazPravnihLica()** – ispisuje na standardni izlaz svako pravno lice, sortirano prema broju korisnika koje pravno lice sadrži (praksuje se ime pravnog lica i broj korisnika)

**dodajFizickoLice(Korisnik korisnik)** – dodaje novog korisnika kao fizičko lice, pri čemu vodi računa da se ne može isti korisnik dodati više puta, a ukoliko se dodaje postpaid korisnik, njegov tarifni paket ne sme biti biznis (biznis atribut ne sme biti true)

### **Prepaid Korisnik:**

**obaviRazgovor(Korisnik korisnik, int brojMinuta)** – cenu razgovora računa kao brojMinuta puta cenaMinuta (koju dobija iz objekta klase MobilniOperater kojem pripada). Ukoliko korisnik ima dovoljno kredita, kredit mu se umanjuje za cenu i metoda vraća true

**posaljiPoruku(Korisnik korisnik, String tekstPoruke)** – ukoliko korisnik koji šalje poruku ima dovoljno kredita (cena poruke je definisana u objektu mobilniOperater kojem korisnik pripada), poruka se šalje, kredit se umanjuje za cenu poruke, a metoda vraća true. U suprotnom metoda vraća false

**surfujNaInternetu(String urlAdresa, int brojMegabajta)** – računa cenu sesije kao brojMegabajta puta cenaMb (iz klase MobilniOperater kojem korisnik pripada) i korisniku umanjuje kredit za izračunati iznos, ukoliko korisnik ima dovoljno kredita i tada metoda vraća true

### **Postpaid Korisnik:**

**obaviRazgovor(Korisnik korisnik, int brojMinuta)** – umanjuje korisniku broj minuta iz tarifnog paketa koji mu pripada (atribut tarifniPaket). Ukoliko brojMinuta premašuje brojMinuta iz tarifnog paketa, računa cenu te razlike (na isti način kao i u kod prepaid korisnika) i za taj iznos uvećava atribut stanje

**posaljiPoruku(Korisnik korisnik, String tekstPoruke)** – najpre proverava da li korisnik ima u svojoj kolekciji neki tarifni dodatak (lista tarifniDodaci) koji je po tipu SMS\_DODATAK i u tom slučaju ne troši poruke iz tarifnog paketa. Ukoliko korisnik nema takav tarifni dodatak, korisniku se umanjuje broj sms poruka iz tarifnog paketa za 1, a ukoliko

je korisnik prethodno potrošio sve poruke iz paketa, atribut stanje mu se uvećava za cenu poruke koja je definisana u mobilnom operateru kojem korisnik pripada

**surfujNaInternetu(String urlAdresa, int brojMegabajta)** – najpre proverava da li postoji tarifni dodatak tipa INTERNET\_DODATAK i u tom slučaju uspešno pravi Internet sesiju i ne troši megabajte iz tarifnog paketa. Ukoliko korisnik nema dodatak, najpre mu se troše megabajti iz paketa (umanjuje se broj megabajta za vrednost koja je prosleđena), a ukoliko ih nema dovoljno, razlika mu se dodatno naplaćuje množeći se sa cenom megabajta

**dodajTarifniDodatak(TarifniDodatak tarifniDodatak)** – dodaje novi tarifni dodatak u listu dodataka, a atribut stanje uvećava za cenu tog dodatka

**generisiRacun()** – u okviru Stringa koji se vraća potrebno je prikazati kao odvojene stavke fiksnu cenu tarifnog paketa, vrednost prekoračenja (stanje atribut), kao i ukupan iznos (koji vraća metoda ukupnoZaNaplatsu())

**ukupnoZaNaplatsu()** – u ukupnu cenu za naplatu ulazi fiksna cena tarifnog paketa, kao i potrošeni iznos van tarifnog paketa (atribut stanje – prekoračenje saobraćaja i cene svih kupljenih tarifnih dodataka)

## **Pravno Lice:**

**dodajKorisnika(PostpaidKorisnik korisnik)** – dodaje novog postpaid korisnika, pri čemu se jedan korisnik ne može dodati više od jednom, a tarifni paket korisnika koji se dodaje mora biti biznis (biznis atribut mora biti true)

**generisiRacun()** – ukupan iznos se računa kao zbir svih računa svakog korisnika koji pripada pravnom licu. Metoda vraća u okviru Stringa naziv pravnog lica, broj korisnika i ukupan iznos računa