

OBJEKTNO ORIJENTISANO PROGRAMIRANJE**- domaći zadatak broj 2 -****Funkcionalna specifikacija**

Na programskom jeziku C++ implementirati telefonski imenik (*Phonebook*).

Učitavanje telefonskog imenika iz fajla

Implementirati operaciju `void Phonebook::loadPhonebook(const string& filepath);` koja iz .csv fajla koji se nalazi na putanji `filepath` učitava telefonski imenik. Ukoliko je prethodno već učitani telefonski imenik, sve informacije o starom imeniku se brišu i učitava se novi imenik. Format fajla je dat u nastavku.

- Fajl se sastoji iz proizvoljnog broja redova. Svaki red sadrži opis jednog kontakta u imeniku.
- Kontakt se sastoji iz imena i broja telefona koji su odvojeni zarezom.
- Ime se sastoji od slova, cifara i znakova razmaka i mora biti jedinstveno u imeniku.
- Broj telefona je napisan u obliku *prefiks broj*, gde se broj sastoji od šest ili sedam cifara, a prefiks je napisan u jednom od dva formata: +381 xx ili 0xx. Razlikuju se mobilni i fiksni telefoni. Mobilni telefoni u prefiksu mogu da imaju vrednosti od 60 do 69, dok fiksni brojevi imaju vrednost 11.

Komande za rad sa telefonskim imenikom

Rad sa telefonskim imenikom se obavlja preko terminala. Terminal podržava komande čiji su opisi dati u tabeli ispod. Predvideti i maksimalno pojednostaviti dodavanje novih komande.

naziv komande	opis
NEW	Dodaje novi kontakt u imenik. Od korisnika se očekuje unos imena i broja telefona preko tastature.
SEARCH	Prikazuje sve kontakte u imeniku čija imena počinju unetim prefiksom. Prikazani kontakti su uređeni leksikografski. Od korisnika se očekuje unos nekoliko početnih slova imena preko tastature. Ukoliko je unos prazan tekst, prikazati sve kontakte.
SELECT	Bira kontakt u imeniku. Od korisnika se očekuje unos imena kontakta preko tastature. Ukoliko kontakt sa unetim imenom ne postoji, prijaviti grešku.
DELETE	Briše odabrani kontakt iz imenika. Ukoliko nijedan kontakt nije odabran komandom SELECT, komanda DELETE nema efekat.
EDIT	Menja podatke odabranog kontakta iz imenika. Ukoliko nijedan kontakt nije odabran komandom SELECT, komanda EDIT nema efekat. Od korisnika se očekuje da odabere deo kontakta koji menja, nakon čega se ime i/ili broj telefona unosi preko tastature.
CALL	Poziva odabrani kontakt iz imenika. Ukoliko nijedan kontakt nije odabran komandom SELECT, komanda CALL nema efekat. Pozvani kontakt se pamti u istoriji poziva.
CALL_HISTORY	Prikazuje kontakte koji su učestvovali u poslednjih deset poziva.

Izvršavanje naredbi preko terminala

Implementirati operaciju `void Phonebook::openTerminal()`; koja otvara interaktivni terminal telefonskog imenika. Terminal funkcioniše tako što ciklično traži od korisnika unos komande. Terminal se zatvara unosom komande **EXIT**.

U nastavku je dat primer rada preko terminala:

```
-----  
Welcome to your phonebook!  
-----  
Enter the command: SEARCH  
- Enter the contact name or first few letters: Mil  
- 3 contacts have been found:  
--- Milan +381 63 1582356  
--- Mileva 011 5486236  
--- Milos Fakultet 011 145663  
Enter the command: SELECT  
- Enter the contact name: Milan  
- Contact Milan +381 63 1582356 is now selected.  
Enter the command: DELETE  
- Contact Milan +381 63 1582356 has been deleted.  
Enter the command: SELECT  
- Enter the contact name: Milan  
- No contact has been found.  
Enter the command: EXIT
```

Obrada grešaka

Neregularne situacije obrađivati konceptom izuzetaka. Sve greške treba izvoditi iz jedne zajedničke klase izuzetaka *PhonebookException*. Prilikom obrade greške, ispisati grešku na standardni izlaz. Ispis treba što bolje da opisuje grešku.

Test funkcija

Javni test je predstavljen fajlom „tst.h“ koji testira funkcionalnost terminala. Studentima je javno dostupna implementacija svih funkcija u datom fajlu i mogu da menjaju postojeće funkcije ili da dodaju nove funkcije da bi dodatno testirali svoj kod na način koji će biti objašnjen na auditornim vežbama. Glavna funkcija se nalazi u fajlu „source.cpp“ i redom poziva sve funkcije iz fajla sa testovima. Ispisi su pokaznog karaktera i mogu biti promenjeni.

Tehnički zahtevi i smernice za izradu rešenja

Programski sistem realizovati tako da bude maksimalno efikasan, detaljno komentarisano, modularan i lako proširiv novim klasama i operacijama. Klasa *Phonebook* i njene operacije moraju biti imenovane prema zahtevima iz domaćeg zadatka. Iz kolekcije standardne biblioteke dozvoljeno je koristiti sledeće tipove: `<string>`, `<vector>`, `<list>`, `<stack>`, `<queue>`. Ukoliko u zadatku nešto nije dovoljno jasno definisano, treba usvojiti razumnu pretpostavku i na temeljima te pretpostavke nastaviti izgrađivanje svog rešenja.