

## PROGRAMSKI JEZICI I – 2. kolokvijum (03.02.2012)

1. (20 bodova) Implementirati hijerarhiju klasa prikazanu na dijagramu klasa (Sl. 1).

Klasa **Sifra** je apstraktna klasa, a služi za šifrovanje i dešifrovanje podataka. Šifrovanje i dešifrovanje podataka vrši se istim ključem. Svaki objekat klase Sifra sadrži informaciju o *modu* u kom se nalazi (šifrovanje ili dešifrovanje). Klasa nema podrazumevani konstruktor, ali je omogućena inicijalizacija modom. Potrebno je preklopiti **operator** `<<` koji (u zavisnosti od konkretne klase) na zadanom izlazu ispisuje naziv algoritma i ključ, a čija se funkcionalnost oslanja na čistu virtuelnu metodu **print**. Preklopiti **operator** `()`, čija je operatorska funkcija čista virtuelna metoda koja vraća šifrovani/dešifrovani podatak na osnovu poruke/šifrata (šifrat – šifrovana poruka) koji je argument ove funkcije.

Klasa **Rotacija** je konkretna klasa izvedena iz klase Sifra. Klasa Rotacija nema podrazumevani konstruktor, ali je omogućena inicijalizacija modom rada i ključem koji se ne može naknadno promeniti.

Ključ Rotacije je prirodan broj ( $n$ ). Poruke se šifruju tako što se svako slovo poruke zameni slovom koje je za  $n$  pozicija pomerenom udesno (npr.  $n=3$ :  $A \rightarrow D$ ,  $B \rightarrow E$ , ...,  $X \rightarrow A$ ,  $Y \rightarrow B$ ,  $Z \rightarrow C$ ,  $a \rightarrow d$ ,  $b \rightarrow e$ , ...,  $z \rightarrow c$ ). Ostali znakovi poruke se prenose bez promene. Poruke se dešifruju tako što se svako slovo šifrata zameni slovom koje je za  $n$  pozicija pomerenom ulevo.

2. (23 boda) Definirati generičku klasu **Niz** koja omogućava manipulaciju nizom podataka nekog tipa. Podaci koji pripadaju nizu treba da se drže u dinamičkom nizu. Parametar šablona je tip podataka ( $T$ ) koji se drži u nizu. Šablon treba da ima:

- **konstruktor**, koji kreira novi (prazan) niz. Konstruktor treba da alocira potreban memorijski prostor na osnovu vrednosti stvarnog argumenta šablona ( $N$ ), a koji predstavlja inicijalni kapacitet niza tj. maksimalni mogući broj elemenata u inicijalnom dinamičkom nizu.
- **operator** `+=`, koji dodaje novi objekat na kraj niza. Ako je niz napunjen podići izuzetak koji se obrađuje u kodu u kojem se operator `+=` koristi.
- **unarni operator** `+`, kojim se povećava kapacitet niza za novih  $N$  elemenata.
- **destruktor**, koji omogućava odgovarajuću destrukciju niza.

3. (12 bodova) Definirati generičku klasu **Iterator** koja omogućava pristup elementima **Niza**. Prilikom inicijalizacije iterator treba da pokazuje na početak niza. Ukoliko u nizu nema elemenata podići izuzetak koji se obrađuje u kodu u kojem se iterator kreira.

Preklapanjem **postfiksno operatora inkrementovanja** omogućiti prelazak sa tekućeg elementa na sledeći element u nizu, i vratiti informaciju o tome da li je prelazak na sledeći element uspešan.

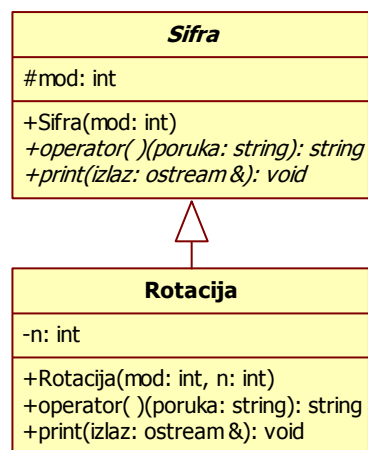
Preklapanjem **unarnog operatora** `*` omogućiti pristup tekućem elementu niza.

4. (15 bodova) U glavnom programu treba:

- iz datoteke, čiji je naziv prvi argument komandne linije prilikom poziva programa, pročitati kompletan sadržaj red po red i napuniti odgovarajući niz stringova.
- sa standardnog ulaza učitati ključ za šifrovanje ( $n$ ).
- u datoteku, čiji je naziv drugi argument komandne linije prilikom poziva programa, upisati šifrate sadržaja datoteke učitane u prvom delu zadatka. Pri šifrovanju koristiti princip dinamičkog vezivanja.

**Napomene:** (1) Za sve klase razdvojiti interfejs od implementacije, tj. funkcije članice implementirati izvan definicije klase. (2) Izuzetke treba podizati pomoću odgovarajućeg objekta.

Sl. 1 Dijagram klasa



Ilustracija uz zadatak 1.

Ključ ( $n$ ): 3

Poruka: PROGRAMSKI JEZICI 1

Šifrat: SURJUDEVNL MHCLFL 1