

## ZADATAK 1

C++

Napisati klasu **Kocka** koja modeluje kocku za igru.

### IMPLEMENTIRATI

- prazan konstruktor
- konstruktor sa parametrima
- konstruktor kopije
- metodu **void baci()**
- metodu **int getVrednost() const**
- slobodnu funkciju **int meni()**

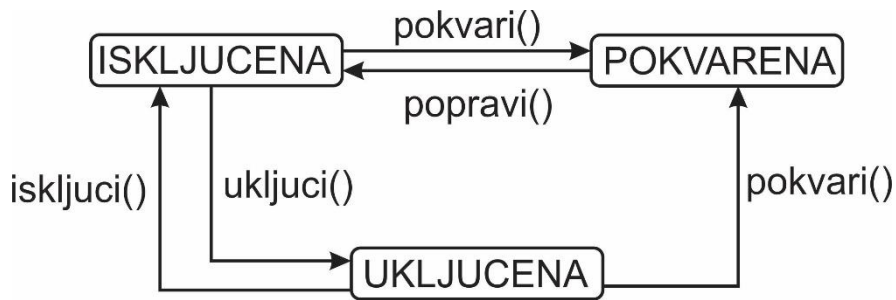
### NAPOMENA

- prazan konstruktor podešava stranu kocke na 1
- za generisanje slučajne vrednosti iskoristiti **rand()** funkciju koja se nalazi u C++ zaglavlju **cstdlib**. Formula za generisanje vrednosti je:  
**rand() % (X-Y+1) + Y**, gde X predstavlja max, a Y min (max = 6, min = 1)
- slobodna funkcija **int meni()** prikazuje listu operacija i kao povratnu vrednost vraća broj izabrane operacije. Na osnovu vraćene vrednosti poziva se odgovarajuća metoda klase.

## ZADATAK 2

C++

Napisati klasu **Rerna** koju opisuje dijagram sa slike.



Klasa treba da sadrži polja: **trenutnoStanje** (nabrojivi tip **StanjeRerne**) i **temperatura** (celobrojni tip).

Temperatura se nalazi u opsegu 0 i 220 °C. Promena temperature se uvek vrši za korak od 20 °C. Temperatura se može menjati samo u stanju UKLJUCENA. U stanju POKVARENA i ISKLJUCENA temperatura ima vrednost 0. Prilikom paljenja rerne, potrebno je automatski povećati temperaturu za jedan korak.

### IMPLEMENTIRATI

- konstruktor bez parametara
- metodu **bool ukljuci()**
- metodu **bool iskljuci()**
- metodu **bool pokvari()**
- metodu **bool popravi()**
- metodu **bool pojacajTemperaturu()**
- metodu **bool smanjiTemperaturu()**
- metodu **StanjeRerne getTrenutnoStanje() const**

- metodu **int getTemperatura() const**
- slobodnu funkciju **void ispisiRernu(const Rerna&)**
- slobodnu funkciju **int meni()**

## **NAPOMENA**

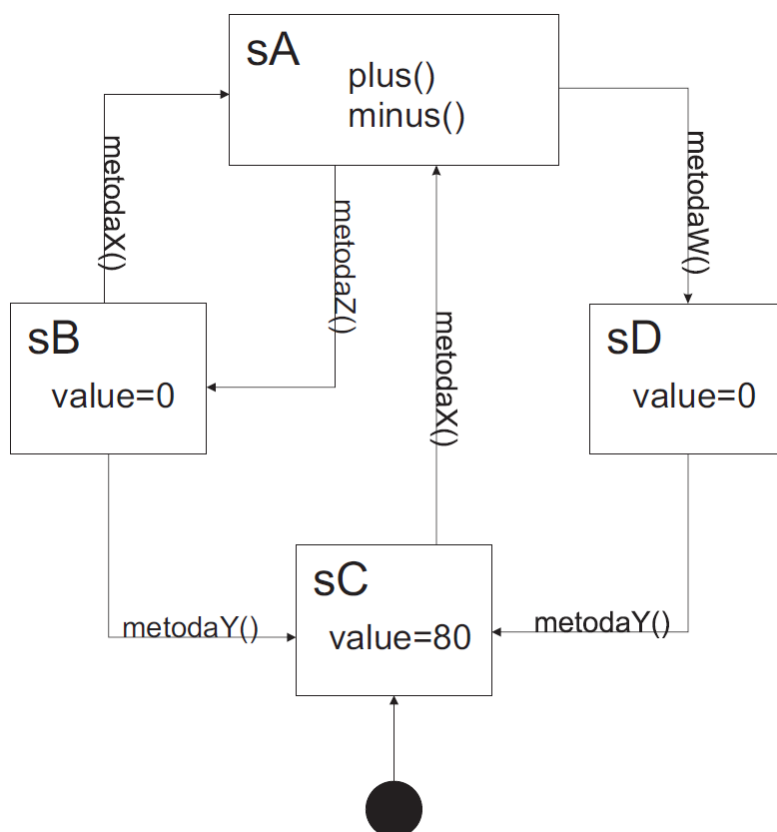
- konstruktor bez parametara postavlja rernu u inicijalno stanje u kom je rerna isključena i njena temperatura je 0
- u realizaciji metoda voditi računa o uslovima i granicama za polje temperatura

## ZADATAK 2-1

C++

Napisati klasu **Machine** koju opisuje dijagram sa slike.

Klasa treba da sadrži polja: **currentState** i **value**. Polje **value** može imati vrednosti između 0 i 80, a menja se sa korakom koji iznosi 20.



### IMPLEMENTIRATI

- konstruktor bez parametara
- metodu **bool metodaX()**
- metodu **bool metodaY()**
- metodu **bool metodaZ()**
- metodu **bool metodaW()**
- metodu **bool plus()**

- metodu **bool minus()**
- metodu **MachineState getCurrentState() const**
- metodu **int getValue() const**
- slobodnu funkciju **void printInfo(const Machine&)**
- slobodnu funkciju **int meni()**

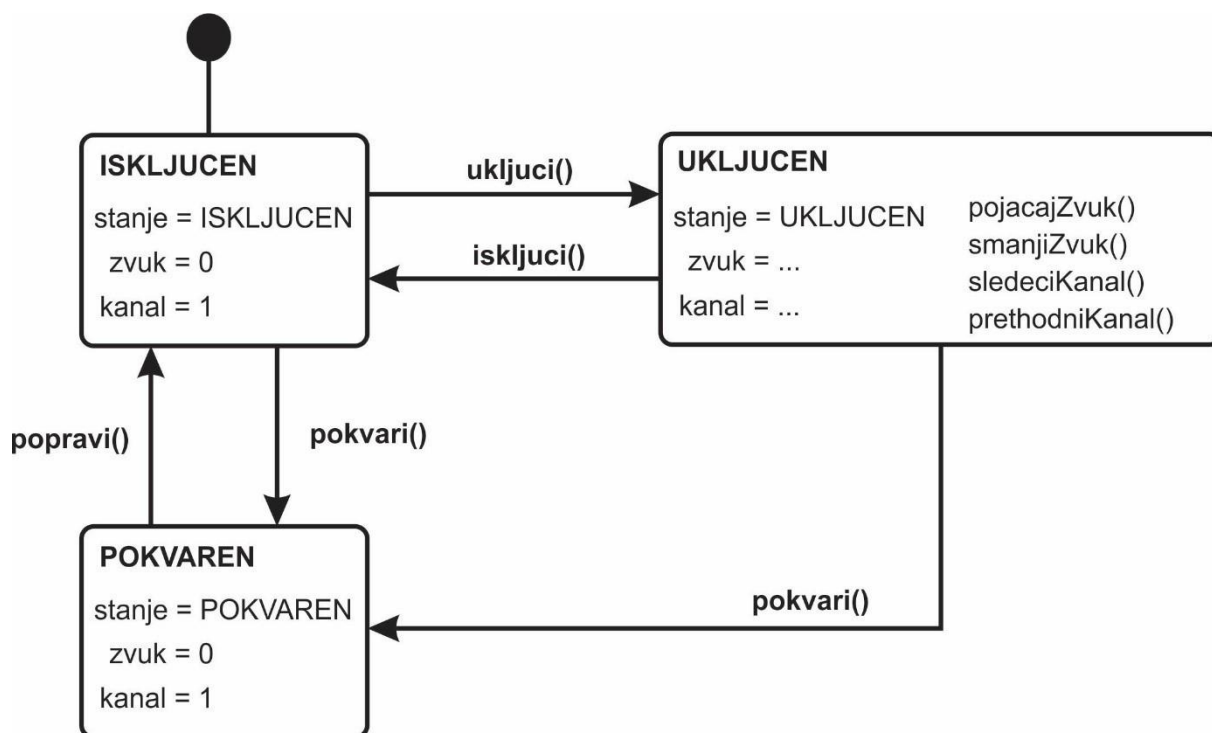
## NAPOMENE

- konstruktor bez parametara mašinu u inicijalno stanje, odnosno u stanje sC sa vrednosti 80
- metoda **bool plus()** povećava vrednost polja value za jedan korak, ukoliko su zadovoljeni uslovi
- metoda **bool minus()** smanjuje vrednost polja value za jedan korak, ukoliko su zadovoljeni uslovi
- u realizaciji metoda voditi računa o uslovima i granicama za polje value

## ZADATAK 2-2

C++

Napisati klasu **Televizor** koju opisuje dijagram sa slike. Klasa treba da sadrži: tri polja (stanje, zvuk i kanal) i odgovarajuće konstruktore i metode.



Polje **stanje** predstavlja stanje televizora i nabrojivog je tipa **StanjeTelevizora**. Objekat se može naći u jednom od sledećih stanja: **UKLJUCEN**, **ISKLJUCEN** ili **POKVAREN**.

Polje **zvuk** predstavlja jačinu zvuka i celobrojnog je tipa. Kreće se u opsegu od **0** do **20**. Promena zvuka se vrši korakom **1**. Korisniku ne treba dozvoliti da izađe iz dozvoljenog opsega.

Polje **kanal** predstavlja kanal koji gledamo, celobrojnog je tipa i kreće se u opsegu od **1** do **5**. Promena kanala se vrši korakom **1**. Korisniku ne treba dozvoliti da izađe iz dozvoljenog opsega.

Zvuk i kanal se mogu promeniti samo kada se objekat nalazi u stanju **UKLJUCEN**. U stanjima **ISKLJUCEN** i **POKVAREN** jačina zvuka treba da bude **0**, a kanal **1**.

Inicijalno stanje je stanje **ISKLJUCEN**.

## IMPLEMENTIRATI

Konstruktor bez parametara i konstruktor kopije

Metode:

- bool ukljuci()
- bool iskljuci()
- bool pokvari()
- bool popravi()
  
- bool pojacajZvuk()
- bool smanjiZvuk()
- bool sledeciKanal()
- bool prethodniKanal()
  
- StanjeTelevizora getStanje() const
- int getZvuk() const
- int getKanal() const

Slobodnu funkciju

- void ispis(const Televizor&)

## NAPOMENA

Svi implementirani konstruktori, metode i slobodne funkcije moraju biti testirani u main() funkciji.

Za potrebe testiranje se može implementirati funkcija meni(), ali nije obavezno.

## ZADATAK 3

C++

Napisati klasu **XnaY** koja modeluje matematičku operaciju  $x^y$ .

### IMPLEMENTIRATI

- konstruktor bez parametara
- konstruktor sa parametrima
- konstruktor kopije
  
- metodu **void setX(int)**
- metodu **void setY(int)**
- metodu **void setX\_Y(int, int)**
  
- metodu **int getX() const**
- metodu **int getY() const**
- metodu **double getXnaY() const**

### NAPOMENE

- konstruktor bez parametara postavlja x i y na 1
- za računanje koristiti metodu **pow()** koja se nalazi u C++ zaglavlju **cmath**.