

# Zadaća 1

Rok za završetak: 14. 11. 2019 u 23:59

Ukupno bodova: 23

Potrebno je da pratite sljedeći link:

<https://classroom.github.com/a/hA7QQbxa>

Na repozitoriju se nalazi projekat koji sadrži samo Main klasu sa Hello World programom, te određeni broj testova. Napišite sve potrebne klase tako da projekat realizira zadatak opisan ispod na način da postojeći testovi prolaze bez izmjene. Također dodajte nove testove tako da pokrivenost koda testovima bude što bliža 100% te da projekat bude optimalno testiran.

## Zadatak 1

Potrebno je napraviti neki osnovni program koji realizira igru šaha. Program neće imati razrađen korisnički interfejs, pošto to nije fokus u ovom trenutku. Pošto nismo radili vještačku inteligenciju, neće biti ni moguće igrati protiv računara. Zanimarićemo i neka od pravila same igre kako bi zadatak bio jednostavniji, kao što su: rokada, razmjena pijuna za druge figure, mat itd.

Pravila igre šah se smatraju da su poznata, ako nisu predlažem da se upoznate koristeći internet.

Program treba sadržavati sljedeće klase:

- Apstraktnu klasu **ChessPiece** i iz nje izvedene klase: **King**, **Queen**, **Rook**, **Bishop**, **Knight** i **Pawn**. Apstraktna klasa treba sadržavati sljedeće metode (čije implementacije ćete eventualno prilagoditi u izvedenim klasama):
  - **getPosition()** vraća String koji predstavlja poziciju figure na tabli u obliku uobičajenom za šah npr. "A4", "B5", "H1" i sl.
  - **getColor()** vraća boju figure; pobrojani tip Color sa vrijednostima BLACK i WHITE treba biti javni statički član klase Chesspiece;
  - **move(String position)** pomjera figuru na novu poziciju; treba baciti izuzetak `IllegalArgumentException` ako je pozicija van table ili nije u ispravnom formatu; ako potez nije dozvoljen za tu vrstu figure (za sada zanemarujemo ostale figure na ploči) treba baciti izuzetak tipa `IllegalChessMoveException`; metoda treba raditi sa velikim i malim slovima;
  - konstruktor klase treba imati parametre pozicija i boja; treba baciti izuzetak `IllegalArgumentException` ako je pozicija van table ili nije u ispravnom formatu.

- Klasa **Board** predstavlja šahovsku ploču sa figurama i u sebi drži sve trenutno aktivne figure na ploči:
  - konstruktor postavlja ploču u uobičajeno početno stanje za igru;
  - **void move(Class type, ChessPiece.Color color, String position)** pomjera figuru koja pripada klasi type, boje color, na poziciju datu stringom position. Ova metoda treba najprije pronaći figuru tipa **type** date boje među aktivnim figurama. Zatim treba pozvati njenu metodu **move** da provjeri da li je poziv legalan. Pošto može biti više figura iste boje i tipa, treba pronaći prvu za koju je potez legalan i povući potez. U slučaju da ne postoji niti jedna figura za koju je potez legalan treba baciti **IllegalChessMoveException**. Konačno treba provjeriti da li se na određenoj poziciji već nalazi figura. Ako se nalazi figura druge boje, ona je "pojeden" i treba je izbaciti, a ako se nalazi figura iste boje treba baciti **IllegalChessMoveException**. Također treba voditi računa da kraljica, lovac, top i pijuni ne mogu preskakati druge figure.
  - **void move(String oldPosition, String newPosition)** još jedna metoda za pomjeranje figure, s tim što su parametri sada stara i nova pozicija figure. Treba napraviti sve provjere kao i u prošloj metodi, ali dodatno ako se na poziciji oldPosition ne nalazi niti jedna figura treba baciti izuzetak **IllegalArgumentException**. Obje metode trebaju raditi sa velikim i malim slovima.
  - **boolean isCheck(ChessPiece.Color color)** provjerava da li je igrač sa bojom color u šahu.
  - *Sve navedene metode moraju biti realizirane pozivanjem metode **move** iz klase ChessPiece! Rješenja koja ne bude urađena na taj način će izgubiti značajan dio bodova!*
- Klasa **Program** sadrži metodu **main** koja omogućuje igračima da povlače poteze. Korisnik unosi naizmjenično poteze bijelog i crnog igrača koristeći skraćenu (algebarsku) notaciju uz oznake figura na engleskom jeziku: K, Q, R, B i N, dok se polja označavaju malim slovom. U metodi main treba biti kreiran objekat tipa Board te koristiti prvu varijantu metode move za povlačenje poteza. U slučaju neispravnog poteza ispisati poruku "Illegal move" i tražiti ponovni unos. U slučaju da je nakon nekog poteza došlo do šaha treba ispisati poruku "CHECK!!!" što se saznaje pomoću metode isCheck. Korisnik može unijeti slovo X da preda partiju. Primjer ulaza i izlaza:
 

```

White move: e4
Black move: e5
White move: Nf3
Black move: Nc6
White move: Bb5
Black move: a6
White move: Ba4
Black move: Nf6
      
```

U zadaći je obavezno pridržavati se svih pravila koja su do sada obrađena na predmetu, te će po tim pravilima biti i ocjenjivano. To uključuje:

- ispravno imenovanje paketa, klasa, metoda i promjenljivih,
- primjenu komentara,
- atomic commits.

Ocjenjivaće se i kvaliteta osmišljenog objektno-orientisanog dizajna kako bi rješenje bilo što jednostavnije i efikasnije za realizaciju.

Sve što nije definisano zadatkom je dozvoljeno. Dakle, možete imati pomoćne klase, metode, osmisлити attribute kako želite itd. itd. baš kao i na svim predmetima do sada, s tim što bi te klase, metode i atributi trebali biti privatni.

Izmjene:

-