

K1

C++

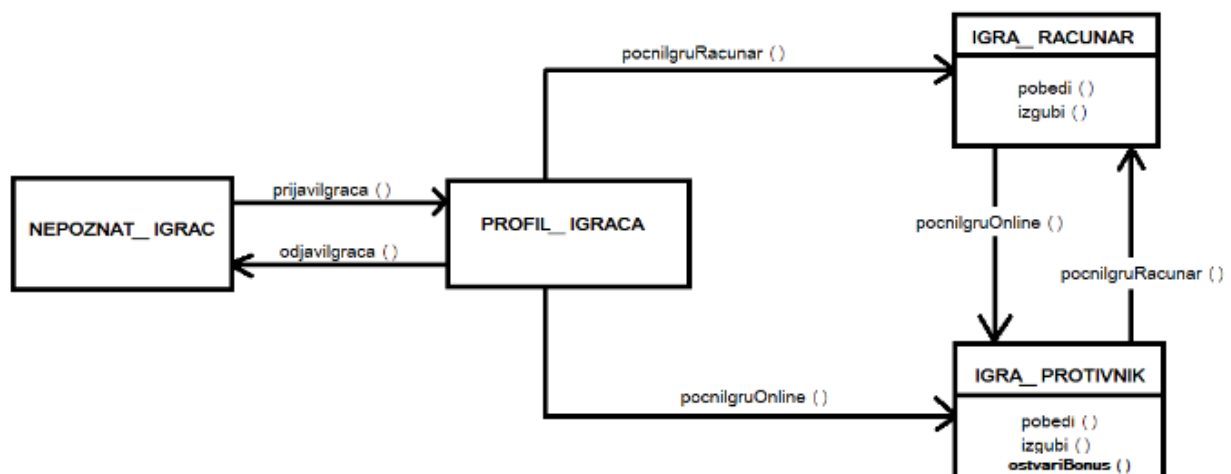
Implementirati klase **Igrac** i **Igra**.

Klasa **Igrac** ima dva polja tipa *int*: **poeni** i **brojPartija**. Za klasu implementirati:

- **Konstruktor** sa podrazumevanim vrednostima za parametre - podrazumevana vrednost za poene je 0, podrazumevana vrednost za broj partija je 0.
- **Konstruktor** kopije
- **get** metode za oba polja
- **set** metodu za polje poeni
- **void dodajPoene(int x)** - dodaje igraču broj poena koji je prosleđen kao parametar
- **void odigrajPartiju()** - povećava broj partija za 1
- **void resetuj()** – postavlja broj odigranih partija i broj poena na 0
- metodu **void ispis() const** - ispisuje sve informacije o igraču

Klasa **Igra** modeluje proces ostvarivanja bodova prilikom igranja igrice na računaru. Klasa **Igra** ima polja: **igrac** (Igrac) i **trenutnoStanje** (nabrojivi tip **StanjeIgre** sa vrednostima: **NEPOZNAT_IGRAC**, **PROFIL_IGRACA**, **IGRA_RACUNAR**, **IGRA_PROTIVNIK**).

Nakon prijave, korisniku se prikazuje njegov profil i nakon toga može da bira da li želi da vežba igrajući protiv računara ili želi da igra protiv nekog protivnika online. Svaki put kad igra protiv računara igrač može da dobije ili izgubi **1** poen, dok kad igra online dobija ili gubi po **2** poena. Broj poena ne može pasti ispod **0**. Maksimalan broj poena koji igrač može da ima u jednom trenutku je **10**. Svaki put kada krene da igra igru, bilo protiv računara ili online, povećava se broj odigranih partija. Broj odigranih partija ne može biti veći od **5**.



Za klasu **Igra** implementirati:

- **Konstruktor** bez parametara koji postavlja stanje na NEPOZNAT_IGRAC i igrača na igrača sa 0 poena i 0 partija.
- **Konstruktor** sa parametrima **Igra(int p, int bp, Stanjelgre si)** - prvi parametar za broj poena, drugi parametar za broj partija i treći parametar za stanje igre
- **bool prijavilgraca()**
- **bool odjavilgraca()** - osim promene stanja resetuje i rezultate
- **bool pocnilgruRacunar()** – ukoliko su ispunjeni uslovi i ukoliko je moguće povećava broj partija igrača i u tom slučaju menja stanje, u suprotnom vraća *false*
- **bool pocnilgruOnline()** – ukoliko su ispunjeni uslovi i ukoliko je moguće povećava broj partija igrača i u tom slučaju menja stanje, u suprotnom vraća *false*
- **bool pobedi()** – povećava igraču broj poena ako je to moguće
- **bool izgubi()** – smanjuje broj poena ako je to moguće, u suprotnom vraća *false*.
- **bool ostvariBonus()** - u slučaju da je igrač u stanju IGRA_PROTIVNIK i da je ukupni broj poena koje je ostvario do tada veći od 5, dodeljuje bonus poene igraču koji su slučajan broj između 1 i 3 i ispisuje mu poruku “*Dobili ste dodatnih <<slučajanBrojKojiJeDodat>> poena.*”. U slučaju da se pređe maksimalan dozvoljeni broj poena nakon dodeljivanja bonusa, broj poena će biti postavljen na maksimalni mogući.
- **prijateljsku funkciju** za ispis **void ispisilgru(const Igra&)** koja ispisuje sve informacije o igri uključujući i informacije o igraču.

NAPOMENE:

1. Pomoć: smanjivanje poena je moguće tako što se doda negativan broj poena.
2. Za generisanje slučajne vrednosti iskoristiti *rand()* funkciju koja se nalazi u C++ zaglavlju *cstdlib*.
3. Pretpostaviti da će korisnici postavljati ispravne vrednosti kroz konstruktore i set metode.
4. Za svaku od klasa dovoljno je napraviti samo **njen** .hpp fajl (nije neophodno posebno praviti i .cpp i .hpp).
5. Svaki napisani fajl obavezno mora biti inkludovan unutar odgovarajućeg fajla.
6. U funkcijama ili metodama za ispis polja nabrojivog i logičkog tipa ispisivati rečima.