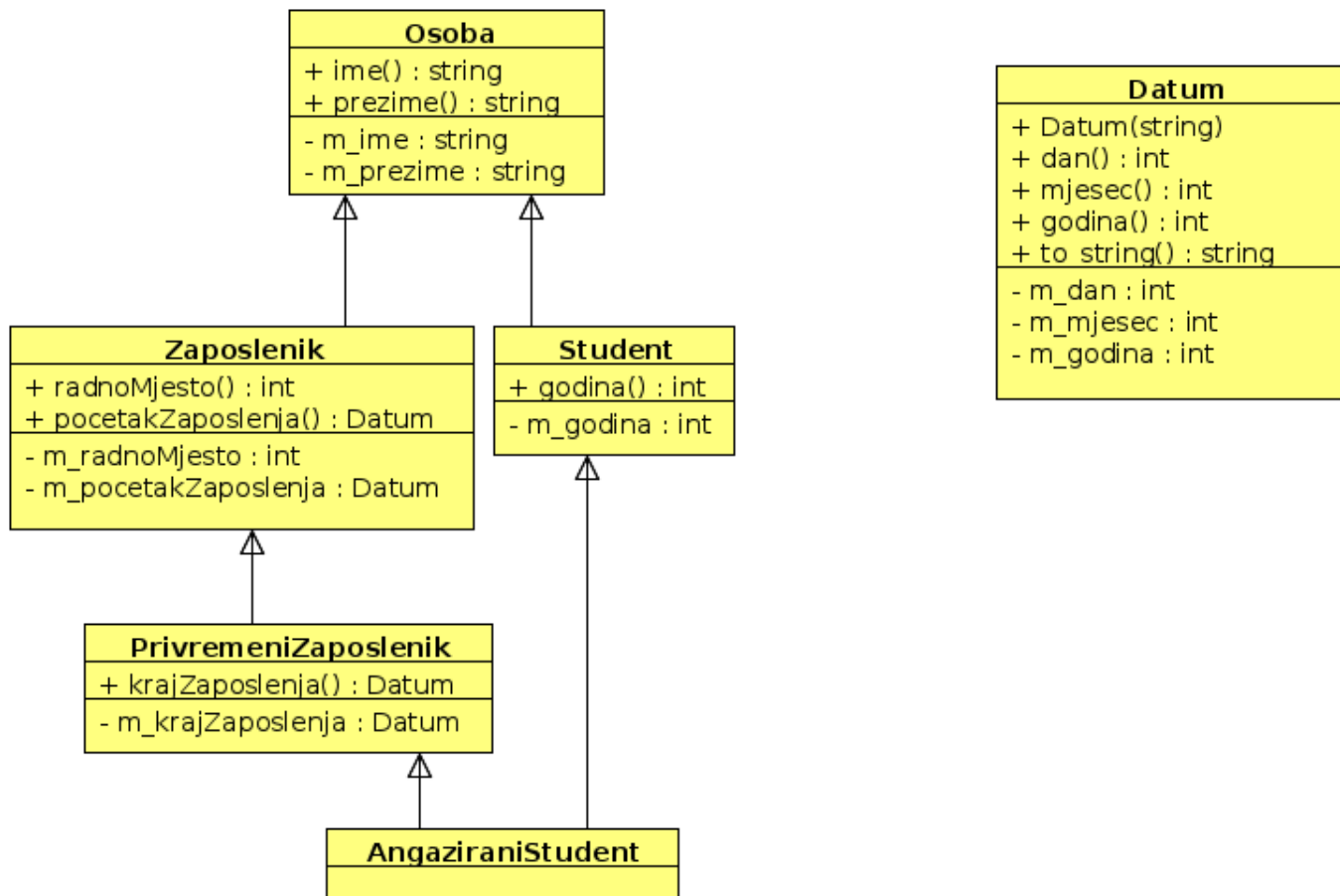


Zadatak

Implementirati stablo nasljeđivanja klasa na slici.



1. Treba ispravno definirati konstruktore. Niti jedna varijabla ne smije ostati neinicijalizirana. Pri tome ne koristiti defaultne vrijednosti. Konstruktore označiti tako da se može vidjeti redosljed konstrukcije podobjekata.
2. **Datum** je pomoćna klasa koja ispisuje datume u obliku "dd:mm:gggg" i može pročitati datum iz stringa tog oblika. Treba implementirati globalni operator za ispis datuma na izlaznom streamu i globalne operatore uspoređivanja datuma.
3. Treba definirati globalne operatore uspoređivanja

```
bool operator==(Osoba const & left, Osoba const & right);
bool operator!=(Osoba const & left, Osoba const & right);
```

koji mogu usporediti bilo koja dva elementa iz stabla nasljeđivanja. Pri tome operator uspoređivanja smije samo jednom zvati na svakom svom argumentu **typeid**. (Uputa: posao uspoređivanja predati funkciji članici oblika

```
bool jednak(Osoba const & o) const;
```

koja treba biti prisutna u svakoj klasi i koja ne provjera tip svog argumenta pa ne smije biti javna.)

4. Napisati globalni operator ispisa

```
std::ostream & operator<<(std::ostream & out, Osoba const & right);
```

koji pravilno ispisuje objekt bilo kojeg tipa iz stabla nasljeđivanja (ispisuje sve podatke, svaki u svojoj liniji). Posebno paziti da se ime i prezime ne ispisuju dva puta.

5. Program testirati sljedećom main-funkcijom:

```
int main()
{
```

```

AngaziraniStudent as1("Ivo", "Josic", 17, "12:02:2015", "12:03:2015", 2);

std::cout << "-----\n";
std::cout << as1 << std::endl;
std::cout << "-----\n";

AngaziraniStudent as2(as1);
assert(as1 == as2);

PrivremeniZaposlenik pz1 = as2;
std::cout << "-----\n";
std::cout << pz1 << std::endl;
std::cout << "-----\n";
assert(pz1 != as2);

AngaziraniStudent * pas3 = new AngaziraniStudent("Nino", "Gusic", 16, "12:02:2015",
"12:03:2015", 2);
Student * ps = pas3;
std::cout << ps->prezime() << std::endl;
std::cout << "-----\n";
std::cout << *pas3 << std::endl;
std::cout << "-----\n";

assert(as2 != *pas3);

std::cout << "-----\n";
delete pas3;
return EXIT_SUCCESS;
}

```

Napomene.

1. Kompilirati s opcijom **-g** (debugging mode) kako bi **assert** naredbe bile uključene u kod.
2. Sve klase staviti u datoteku imena **mi.h**. Sav kod koji implementira funkcije članice te main funkciju staviti u datoteku imena **main.cpp**.