Destruktor

- Ako klasa *X* nema eksplicitno deklarisan destruktor (tj. korisnički definisan destruktor), prevodilac će implicitno deklarisati destruktor koji je javan i *inline*
- Programer može zahtevati automatsko generisanje destruktora koji bi prevodilac implicitno deklarisao specifikatorom *=default*
- Ako iz bilo kog razloga nije moguće uništavanje podobjekata osnovnih klasa i objekata članova (npr. jer su im detruktori nedostupni), destruktor se smatra obrisanim (nije moguć njegov implicitan ili eksplicitan poziv)
- Ako implicitno deklarisan ili podrazumevani destruktor nije obrisan, prevodilac generiše njegovu definiciju sa praznim telom
- Kada se uništava neki objekat, najpre se izvršava telo destruktora, a nakon toga se implicitno pozivaju destruktori objekata članova (ako su objekti klasa) i destruktori osnovnih klasa, uvek po redosledu obrnutom od redosleda inicijalizacije

Destruktor

- Destruktor može biti i virtuelan; destruktor deklarisan kao virtuelan u osnovnoj klasi ostaje virtuelan i u izvedenim klasama:

```
struct Base {
  virtual ~Base ();
};
```

- Tada je poziv destruktora polimorfan u situacijama kada se objektu pristupa posredno, preko pokazivača ili reference na osnovnu klasu:

```
Base* pb = new Derived;
delete pb;
```

- U ovakvoj situaciji, biće pozvan najpre destruktor izvedene klase, koji uvek implicitno poziva i destruktore osnovnih klasa, pa u telu destrkutora nikada ne treba pisati eksplicitan poziv destruktora osnovne kalse
- Ukoliko destruktor ne bi bio virtuelan, u ovakvim situacijama ponašanje bi bilo nedefinisano. Zbog toga se preporučuje da destruktor klase uvek bude virtuelan, ako je klasa polimorfna (ako ima bar jednu virtuelnu funkciju članicu), jer se njenim objektima pristupa polimorfno: računa se na to da se objektu pristupa kao generalizovanom entitetu, bez znanja o konkretnom tipu tog objekta, što važi i za uništavanje
- Destruktor može da se deklariše i kao čisto virtuelan (*=0*), na primer u osnovnoj klasi koja treba da bude apstraktna, a nema drugu pogodnu funkciju članicu koja bi bila čisto virtuelna. Takvi destruktori ipak moraju da imaju definiciju, jer se destruktor osnovne klase uvek poziva kada se uništava objekat izvedene klase