
Ugrađeni objekti

- ❖ Kada se inicijalizuje objekat klase, redosled inicijalizacije je sledeći:
 - ako se radi o najizvedenijem objektu (tj. ovo je “prva” inicijalizacija objekta u hijerarhiji nasleđivanja), inicijalizuju se podobjekti virtuelnih osnovnih klasa, po redosledu obilaska grafa nasleđivanja po dubini, sleva nadesno (po redosledu kako su te osnovne klase navedene u deklaracijama izvedenih klasa)
 - inicijalizuju se podobjekti direktnih osnovnih klasa (iz koje je ova klasa neposredno izvedena), po redosledu njihovog navođenja u deklaraciji izvedene klase
 - inicijalizuju se objekti članovi
 - izvršava se telo konstruktora klase čiji je objekat direktna instanca
- ❖ Treba primetiti da svako pominjanje “inicijalizacije (pod)objekta” u navedenim pravilima znači rekurzivnu inicijalizaciju podobjekata osnovnih klasa, objekata članova i izvršavanje tela konstruktora za taj (pod)objekat
- ❖ Redosled uništavanja (destrukcije) objekta uvek je tačno obrnut:
 - izvršava se telo destruktora klase čiji je objekat direktna instanca
 - uništavaju se objekti članovi, po redosledu obrnutom od redosleda inicijalizacije
 - uništavaju se podobjekti direktnih osnovnih klasa, po redosledu obrnutom od redosleda inicijalizacije
 - ako se radi o najizvedenijem objektu, uništavaju se podobjekti virtuelnih osnovnih klasa, po redosledu obrnutom od redosleda inicijalizacije

Ugrađeni objekti

❖ Na primer:

```
struct D { D(char c) { cout<<"D"<<c; } };  
struct A {  
    D d;  
    A() : d('a') { cout<<"A\n"; }  
};  
struct B : A {  
    D d;  
    B() : d('b') { cout<<"B\n"; }  
};  
struct C {  
    D d;  
    C() : d('c') { cout<<"C\n"; }  
};  
struct X : B, C {  
    D d;  
    D a[3];  
    X() : C(), B(), d('x'), a{'0','1','2'} { cout<<"X"; }  
};  
int main () {  
    X x;  
}
```