

# 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;  
}
```

Pre izvršavanja tela konstruktora izvedene klase, najpre se inicijalizuju podobjekti osnovnih klasa pozivom njihovih konstruktora, pa onda i objekti članovi. Bez obzira na redosled ovde navedenih inicijalizatora, redosled inicijalizacije je uvek određen redosledom navođenja osnovnih klasa i podataka članova u definiciji izvedene klase

Ispis će biti:

*DaA*

*DbB*

*DcC*

*DxD0D1D2X*

Zadatak: odrediti redosled uništavanja, odnosno poziva destruktora; napisati destruktore koji ispisuju slične poruke

---

# Glava 12: Konstruktor, destruktor i operator dodele

---

- ❖ Konstruktor
- ❖ Podrazumevani konstruktor
- ❖ Konstruktor kopije
- ❖ Destruktor
- ❖ Operator dodele kopiranjem

