

Operator dodele kopiranjem

- ❖ *Operator dodele kopiranjem* (*copy assignment operator*) klase *X* je nestatička operatorska funkcija članica te klase sa imenom *operator=* čiji je jedini parametar tipa *X*, *X&*, *const X&*, *volatile X&* ili *const volatile X&*
- ❖ Ova funkcija poziva se kada se objekat klase *X* nalazi kao levi operand operatora dodele *=*, i kada se baš ona odabere u postupku odabira preklopljene operatorske funkcije za desni operand tog operatora:

```
struct X {  
    X& operator= (const X&);  
    ...  
};
```

Operator dodele kopiranjem

Ovde se poziva konstruktor kopije

```
X x1, x2 = x1, x3;  
x3 = x2;
```

Ovde se poziva podrazumevani konstruktor

Ovde se poziva operator dodele kopiranjem: *x3.operator=(x2)*

- ❖ Operacija dodele kopiranjem je potpuno drugačija operacija od inicijalizacije kopiranjem:
 - inicijalizacija kopiranjem se vrši na početku životnog veka objekta, i tada se poziva konstruktor kopije; pre toga objekat koji se inicijalizuje ne postoji
 - dodela kopiranjem se vrši u izrazu, kao operacija dodele, i tada se poziva operator dodele; pre toga objekat kome se dodeljuje već postoji i inicijalizovan je ranije

Operator dodele kopiranjem

- ❖ Ako klasa X nema nijedan eksplicitno deklarisan operator dodele kopiranjem (tj. korisnički definisan takav operator), prevodilac će implicitno deklarirati operator dodele kopiranjem koji je *javan*, *inline*, i za koji važi:
 - ako svaka direktna osnovna klasa B ima operator dodele kopiranjem koji ima parametar tipa B , *const B&* ili *const volatile B&*, i ako svaka klasa M podatka člana ima operator dodele kopiranjem koji prima parametar tipa M , *const M&* ili *const volatile M&*, onda i ovaj implicitno deklarirani operator dodele ima oblik $X\& \text{ operator}=(\text{const } X\&)$
 - u suprotnom, ovaj operator dodele ima oblik $X\& \text{ operator}=(X\&)$
- ❖ Klasa može imati i više operatora dodele kopiranjem, recimo onaj koji prima $X\&$ i onaj koji prima *const X&*
- ❖ Ako klasa ima neki korisnički definisan operator dodele kopiranjem (pa prevodilac ne generiše onaj implicitni), programer ipak može forsirati generisanje podrazumevanog operatora dodele kopiranjem specifikatorom *=default*:

```
struct X {  
    X& operator= (X&);  
    X& operator= (const X&) = default;  
};
```

- ❖ Pošto klasa tako uvek ima deklarisan operator dodele kopiranjem, on uvek sakriva takav operator iz osnovne klase
- ❖ Implicitno deklarisan operator dodele kopiranjem biće smatran obrisanim ako ta klasa ima korisnički definisan konstruktor premeštanja (*move constructor*) ili operator dodele premeštanjem (*move assignment operator*)
- ❖ Ako podobjekti osnovnih klasa i objekti članovi ne mogu da se dodeljuju kopiranjem, recimo zato što su objekti klasa koje imaju nedostupne operatore dodele kopiranjem, ili ako je član konstanta ili referenca, onda će podrazumevani operator dodele kopiranjem biti obrisan (smatraće se da njegov poziv nije dozvoljen, iako je on deklarisan)
- ❖ U suprotnom, ako ovaj implicitno deklarirani ili podrazumevani operator dodele kopiranjem nije obrisan, on će biti definisan i vršiće podrazumevanu dodelu kopiranjem podobjekata osnovnih klasa i objekata članova, po istom redosledu kao i u inicijalizaciji; ako su ti podobjekti objekti nekih klasa, pozivaju se njihovi operatori dodele kopiranjem, u suprotnom, vrši se prosto kopiranje vrednosti