

Inicijalizacija kopiranjem

❖ *Inizijalizacija kopiranjem (copy initialization)* se obavlja kada se objekat (ne referenca) tipa T (koji je objektni tip) inicijalizuje u sledećim slučajevima:

- Kada se imenovani objekat (automatski, statički ili vezan za nit) inicijalizuje izrazom iza znaka `=`:

```
T t = expression;
```

- Kada se argument prenosi u pozvanu funkciju po vrednosti (ne referenci):

```
void f (T t);
```

```
f(expression);
```

- Kada se vraća iz funkcije koja vraća vrednost (ne referencu):

```
T f () {
```

```
...
```

```
    return expression;
```

```
}
```

- Kada se baca ili hvata izuzetak po vrednosti (ne referenci):

```
throw expression;
```

```
catch (T t) {...}
```

- Kao deo agregatne inicijalizacije, za inicijalizaciju svakog elementa za koji je zadat inicijalizator:

```
T a[N] = {expression, expression, ...};
```

Inicijalizacija kopiranjem

❖ Inicijalizacija kopiranjem radi sledeće:

- Ako je T klasa, a izraz kojim se inicijalizuje jeste čdvrednost (*prvalue*) koja je istog tipa T (bez cv-kvalifikacije), onda se izostavlja kopiranje i objekat inicijalizuje onim što ta čdvrednost predstavlja, kao što je ranije objašnjeno (pre verzije C++17, pravio se privremeni objekat kao rezultat izraza koji se premeštao ili kopirao u objekat koji se inicijalizuje, uz moguće izostavljanje kopiranja):

```
X x = X();
```

```
extern X f(...);  
X x = f(...);
```

- Ako je T klasa, a tip izraza kojim se objekat inicijalizuje (bez cv-kvalifikacije) klasa izvedena iz te klase, pretražuju se konstruktori klase T koji nisu *explicit* i koji mogu da prihvate tip izraza kao argument i bira se onaj konstruktor koji najbolje odgovara, po pravilima rezolucije za preklapanje funkcija; objekat se onda inicijalizuje tim konstruktorom:

```
Derived d;
```

```
Base b = d;
```

- Ako je T klasa, a tip izraza kojim se objekat inicijalizuje (bez cv-kvalifikacije) nije klasa izvedena iz te klase, ili T nije klasa, a tip izraza jeste klasa, traže se korisnički definisane konverzije koje mogu da inicijalizuju objekat tipa T iz tipa izraza; ako je odabrana korisnička konverzija zadata konverzionim konstruktorom, objekat se inicijalizuje tim konstruktorom direktno (bez pravljenja privremenog objekta, kao ranije):

```
X x = y;
```

- U preostalim slučajevima, kada ni T ni tip izraza nisu klase, vrednost izraza se prosto kopira u objekat, ako je izraz istog tipa T , ili se po potrebi koriste standardne konverzije da konvertuju vrednost izraza u tip objekta:

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