
Direktna inicijalizacija

❖ *Direktna inicijalizacija* (*direct initialization*) inicijalizuje objekat eksplicitnim skupom argumenata konstruktora u sledećim situacijama:

- Inicijalizacija imenovanog objekta nepraznom listom izraza unutar zagrada:

`T t (expression, ...)`

- Inicijalizacija objekta neklasnog tipa jednim izrazom unutar velikih zagrada:

`T t {expression}`

- Inicijalizacija privremenog objekta kao čdvrednosti nepraznom listom izraza unutar zagrada:

`T (expression, ...)`

- Inicijalizacija privremenog objekta kao čdvrednosti izraza *static_cast*:

`static_cast<T>(expression)`

- Inicijalizacija dinamičkog objekta u izrazu *new* pomoću inicijalizatora sa nepraznim zagradama:

`new T (expression, ...)`

- Inicijalizacija podobjekta osnovne klase ili podobjekta člana unutar liste inicijalizatora u konstrukturu za nepraznom listom unutar zagrada:

`C::C (...), B (expression, ...), m (expression,...) {...}`

Direktna inicijalizacija

❖ Direktna inicijalizacija radi sledeće:

- Ako je T klasa, a jedini 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 parametar konstruktora kojim se objekat inicijalizuje, uz moguće izostavljanje kopiranja):

X x1 (x2);

- Ako je T klasa, a tip izraza kojim se objekat inicijalizuje (bez cv-kvalifikacije) nije jedini ili nije istog tipa, pretražuju se konstruktori klase T koji mogu da prihvate te argumente i bira se onaj konstruktor koji najbolje odgovara, po pravilima rezolucije za preklapanje funkcija; objekat se onda inicijalizuje tim konstruktorom
- Ako T nije klasa, a tip izraza kojim se objekat inicijalizuje (bez cv-kvalifikacije) jeste klasa, traže se korisnički definisane konverzije operatorske funkcije klase izraza ili njene osnovne klase, ako ih ima, i bira se ona koja najbolje odgovara, i objekat tipa T se onda inicijalizuje tom konverzijom
- Inače, ako je T tip *bool*, a izvorni tip *std::nullptr_t* (tip konstante *nullptr*), vrednost se inicijalizuje na *false*
- 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);

- ❖ Direktna inicijalizacija je manje restriktivna i više dozvoljava nego inicijalizacija kopiranjem: inicijalizacija kopiranjem uzima u obzir samo korisnički definisane konverzije (konstruktor konverzije i konverzije funkcije) koje nisu *explicit*, dok direktna inicijalizacija uzima u obzir i njih