

Inicijalizacija nulom

❖ Inicijalizacija nulom radi sledeće:

- Ako je T skalarni tip, inicijalna vrednost objekta je celobrojna konstanta nula eksplicitno konvertovana u tip T ; za pokazivače, to je uvek *null* vrednost, čak i ako se ona ne predstavlja binarnom vrednošću nula
- Ako je T klasni tip, svi podobjekti osnovne klase i članovi se inicijalizuju nulama, a konstruktori se ignorišu
- Ako je T niz, elementi se inicijalizuju nulama
- Za reference se ništa ne radi

static T t;

❖ Na primer:

```
struct X {  
    int m;  
};
```

Kao statički objekti, inicijalizuju se nulama

```
int i;  
int* p;
```

Inicijalizuje se nulama kao deo inicijalizacije vrednošću

```
int main () {  
    X x{};
```

$a[1]$ se inicijalizuje nulom kao deo inicijalizacije vrednošću, jer nije zadat inicijalizator za taj element

```
    int a[2][1];
```

```
    int j{};
```

Inicijalizuje se nulama kao deo inicijalizacije vrednošću, jer je *int* neklasni tip

```
    delete p;
```

Bezbedno je raditi *delete* za pokazivač koji ima *null* vrednost

```
    cout<<x.m<<i<<a[1]<<j;
```

```
}
```

Ispisaće se: 0000

Podrazumevana inicijalizacija

❖ *Podrazumevana inicijalizacija (default initialization)* se obavlja u sledećim slučajevima:

- Kada se varijabla sa automatskim, statičkim ili trajanjem skladišta vezanim za nit deklariše bez inicijalizatora:

```
T t;
```

- Kada se pravi dinamički objekat izrazom *new* bez navedenog inicijalizatora:

```
new T
```

- Kada se osnovna klasa ili nestatički podatak član ne navede u listi inicijalizatora članova u konstruktoru klase, a taj konstruktor se poziva:

```
struct B {...};
```

```
struct D : B {  
    B b;  
    D () {}  
}
```

```
D d;
```

❖ Podrazumevana inicijalizacija radi sledeće:

- Ako je *T* klasni tip, poziva se podrazumevani konstruktor bez argumenata
- Ako je *T* niz, svi elementi se inicijalizuju podrazumevanom inicijalizacijom
- Inače, ne radi se ništa; zbog toga automatski i dinamički objekti (i njihovi podobjekti) neklasnih tipova imaju neodređene vrednosti (šta god se zateklo u memoriji)