

Opseg važenja

❖ Opseg važenja imena počinje od njegove tzv. *tačke deklarisanja* (*point of declaration*), koja se razlikuje za različite entitete, na primer:

- za enumeracije, klase ili šablone, to je odmah iza mesta identifikatora:

```
class X {  
    X (const X&);  
    ...  
};
```

Ovo ime X odnosi se baš na ovu klasu X koja se definiše

- za varijable, to je odmah iza deklaratora, a pre inicijalizatora:

```
const int x = 1, *p = &x;  
{  
    int* x[x] = {x};  
}
```

Ime x već važi, pa je ovo u redu

Ovo ime x odnosi se na ono tipa *const int*,

Ovo ime x odnosi se na ono tipa *int*[1]*, pa prvi element niza ukazuje na samog sebe

Opseg važenja bloka

- ❖ Potencijalni opseg važenja varijable koja je uvedena deklaracijom unutar bloka (složene naredbe) počinje od tačke deklarisanja i završava se na kraju tog bloka. Stvarni opseg važenja je isti kao i potencijalni opseg važenja, osim ako postoji ugnežđeni blok sa deklaracijom koja uvodi identično ime; u takvom slučaju, iz potencijalnog opsega važenja se isključuje taj ugnežđeni blok:

```
void f () {  
    int x = 1;  
  
    x = 2;  
  
    {  
        long x = 0;  
        x = 2;  
    }  
  
    x = 3;  
}
```

int y = x;

- ❖ Potencijalni opseg važenja imena formalnog parametra funkcije počinje od tačke deklarisanja i završava se na kraju cele deklaracije te funkcije
- ❖ Potencijalni opseg važenja imena deklarisanog u hvataču izuzetka (parametar *catch* bloka) počinje od tačke deklarisanja i završava se na kraju tog *catch* bloka, i nije u opsegu važenja u drugom *catch* bloku istog *try* konstrukta