

Nabrajanja (enumeracije)

- ❖ Vrednost koja se dobija ovom konverzijom definisana je na sledeći način:
 - ona koja je eksplicitno zadata inicijalizatorom (koji mora biti konstantni izraz)
 - ako inicijalizator nije zadat, jednaka je vrednosti prethodnog enumeratora plus 1; vrednost prvog enumeratora je podrazumevano 0, ako nije eksplicitno zadat inicijalizator:

```
enum Status { initiated = 1, suspended, //suspended == 2  
             committed, canceled = 5,  
             failed = committed+4 }; // failed == 7
```

- ❖ Suprotna konverzija iz celobrojnog tipa u tip enumeracije može se raditi samo eksplicitno, ne i implicitno; ukoliko data celobrojna vrednost ne spada u skup vrednosti enumeratora, ponašanje je nedefinisano; ovakve konverzije svakako nisu preporučljive:

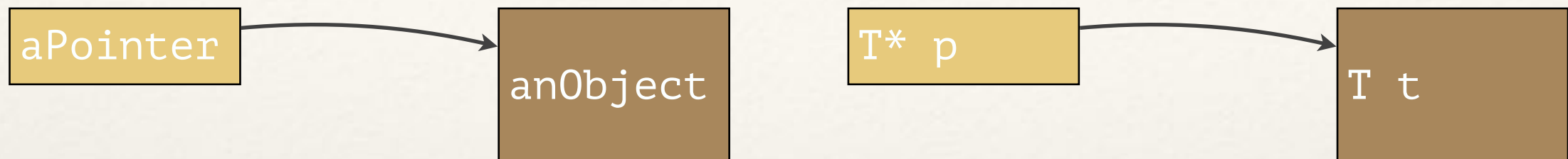
```
Status s = 3;
```

Greška u prevođenju

```
Status s = static_cast<Status>(3);
```

Ovo nije dobar način programiranja!

Pokazivači na objekte



- ❖ Pokazivač (*pointer*) na objekat je objekat koji *ukazuje* (*points to*) na neki drugi objekat; konceptualno, ovo predstavlja *jednosmernu* (*unidirekcionu*, *unidirectional*) vezu pokazivača ka objektu
- ❖ Ako je T neki objektni tip, tip pokazivača na tip T je tip T^* ; pokazivač ovog tipa može se inicijalizovati rezultatom, ili mu se može dodeliti rezultat operatora *uzimanja adrese* (*address-of*) čiji je operand objekat tipa T ; tada pokazivač ukazuje na taj objekat:

```
T t;  
T* p = &t;
```

- ❖ Rezultat operatora *indirekcije* (operator $*$) čiji je operand pokazivač na T odnosi se na objekat na koga taj pokazivač ukazuje:

```
...*p...
```