
Reference

- ❖ Ceo koncept uopšten je na postojanje složenog tipa *reference* (*reference*): referenca je posrednik do objekta, i za vreme svog životnog veka *upućuje* na (referencira, *refers to*) objekat
- ❖ Referenca je posrednik do objekta, slično kao i pokazivač, ali se i značajno razlikuje od pokazivača po sledećem:
 - referenca nije objekat (pokazivač jeste), već je posebna kategorija entiteta u jeziku; zbog toga, recimo, ne postoje reference na reference, pokazivači na reference, niti nizovi referenci (a postoje reference na pokazivače, pokazivači na pokazivače i nizovi pokazivača); ne postoje ni reference na tip *void*
 - referenca se mora vezati za neki objekat na početku svog životnog veka, odnosno mora biti inicijalizovana vezivanjem za objekat; pokazivač ne mora da se veže za objekat
 - referenca ne može da se preusmeri na drugi objekat, dok pokazivač može
 - referenca ne može da se “raskine”, odnosno da se “razveže” od objekta i da ne upućuje ni na jedan objekat, dok pokazivač može (*null* vrednost pokazivača)
- ❖ Referenca se, u principu, implementira isto kao i pokazivač — njena vrednost sadrži adresu referenciranog objekta, s tim da u određenim situacijama, kada to može da se izvede, prevodilac tu vrednost može da zna i za vreme prevođenja, pa vrednost reference ne mora ni da postoji za vreme izvršavanja (iako najčešće postoji), odnosno ta vrednost ne mora da se smešta nigde za vreme izvršavanja, već postoji samo konceptualno

Reference

- ❖ Ne postoji operacija nad samom referencom koja bi joj promenila vrednost, odnosno “prevezala” je na neki drugi objekat nakon inicijalizacije: *svaka* upotreba reference nakon njenog deklarisanja odnosi se na referencirani objekat; zbog toga se referenca često naziva sinonimom za objekat (ukoliko sama referenca ima svoje ime, što ne mora)

- ❖ Na primer:

```
int i = 1;  
int& j = i;  
i=3;  
j=5;  
int* p = &j;  
j+=1;  
int k=j;  
int m=*p;
```

- ❖ Treba dobro razlikovati znak & upotrebljen u deklaraciji reference (sastavni deo tipa), od znaka & upotrebljenog u izrazu, kada označava operaciju uzimanja adrese; ako se u ovom drugom slučaju kao operand nađe referenca, uvek se odnosi na referencirani objekat
- ❖ Pristup do objekta preko reference ne zahteva operator: sama upotreba reference u izrazu odnosi se na referencirani objekat; pristup do objekta preko pokazivača zahteva operator