

# 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* (*adress-of*) čiji je operand objekat tipa  $T$ ; tada pokazivač ukazuje na taj objekat:

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

*p ukazuje na t*

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

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

*\*p predstavlja objekat na koji p ukazuje*

---

# Pokazivači na objekte

---

❖ Na primer:

```
int i=0, j=0;
int* pi = &i;
*pi = 1;
pi = &j;
*pi = 2;
```

❖ Pošto su i pokazivači i nizovi objekti, pokazivač može ukazivati i na njih:

```
int i=0, j=0;
int* pi=&i;
int** ppi;
ppi=&pi;
*pi=1;
**ppi=2;
*ppi=&j;
ppi=&i;

int a[10];
int (*pa)[10] = &a;
```

❖ Ako pokazivač ukazuje na objekat klasnog tipa, može se pristupiti članu tog objekta preko operatora ->; ukoliko se radi o objektu polimorfne klase (sa bar jednom virtuelnom funkcijom), aktivira se polimorfizam:

```
Clock* pc = new ClockWithDates;
pc->tick();
(*pc).tick();
```