

Polimorfizam

- ❖ Izvedena klasa može da dâ svoju definiciju virtuelne funkcije, tj. da je *redefiniše*, ali i ne mora (onda nasleđuje implementaciju):

```
class ClockWithDate : public Clock {  
public:  
    ClockWithDate (int y, int m, int d, int h, int min, int s);  
    virtual void tick ();  
    virtual string getTime ();  
    string getDate ();  
    void setDate (int yy, int mm, int dd);
```

Redefinisane virtuelne funkcije

```
private:
```

```
    int y, m, d;
```

```
};
```

```
void ClockWithDate::tick () {  
    Clock::tick();  
    if (isMidnight()) {  
        // increment the date  
    }  
}
```

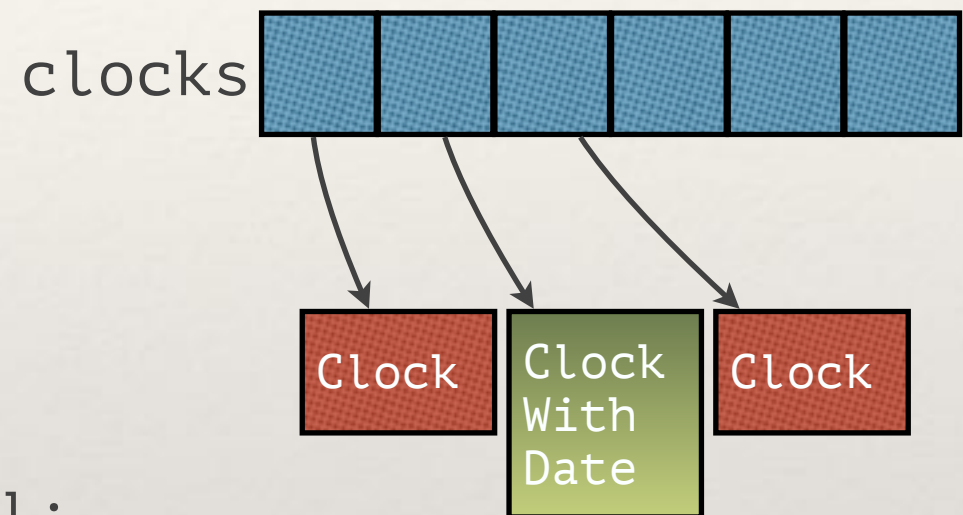
Poziv verzije implementacije iz osnovne klase

```
string ClockWithDate::getTime () {  
    return to_string(m) + "/" + to_string(d) + "/" +  
           to_string(y) + " " + Clock::getTime();  
}
```

Polimorfizam

- ❖ Drugi delovi programa koji objektima pristupaju generalizovano, kao (direktnim ili indirektnim) instancama osnovne klase *Clock*, ne moraju da se menjaju:

```
void Lobby::tick () {  
    for (int i=0; i<num; i++)  
        clocks[i]->tick();  
}  
  
void Lobby::print () {  
    for (int i=0; i<num; i++)  
        cout<< cities[i] << ": "  
            << clocks[i]->getTime() << endl;  
}
```



- ❖ Bez obzira na to što se objektu obraća kao instanci osnovne klase (pokazivač je tipa *Clock**), odaziva se ona implementacija operacije koja odgovara klasi čija je objekat direktna instanca; svaki objekat reaguje na način svojstven svojoj klasi
- ❖ Ovaj mehanizam naziva se *polimorfizam* (*polymorphism*, *poly* - više, *μορφή* - oblik, pojavljivanje u više oblika)