

Poziv funkcije

- ❖ Ukoliko je funkcija virtuelna funkcija članica, poziv funkcije je polimorfan (tj. polimorfizam se aktivira) ako i samo ako se poziv vrši preko pokazivača ili reference na objekat:

```
class Base {  
    public: virtual void f();  
};  
  
class Derived : public Base {  
    public: virtual void f() override;  
};  
  
void g (Base* pb) {  
    pb->f();  
}  
  
void main () {  
    Derived d;  
    g(&d);  
  
    Base b;  
    g(&b);  
}
```

Ovaj poziv je uvek polimorfan; isto je i za poziv preko reference

Poziva se *Derived::f()*

Poziva se *Base::f()*

Poziv funkcije

- ❖ U suprotnom, ukoliko se poziv funkcije članice vrši preko objekta, poziv nije polimorfan:

```
void h (Base b) {  
    b.f();  
}
```

```
void main () {  
    Derived d;  
    h(d);  
  
    Base b;  
    h(b);  
}
```

- ❖ Dinamičko vezivanje je implementaciona tehnika kojom se realizuju polimorfni pozivi. Treba primetiti to da bi u ovakvim pozivima za objekat (ne preko pokazivača ili reference), čak i ako bi taj poziv bio izvršen dinamičkim vezivanjem, rezultat bio isti (jer VTP objekta tipa *Base* ukazuje na VT baš te klase). Za pozive preko pokazivača ili reference, prevodilac, u opštem slučaju, ne može znati šta se krije iza objekta, jer to može biti i promenljivo (npr. za parametre funkcija), pa će te pozive uvek implementirati dinamičkim vezivanjem