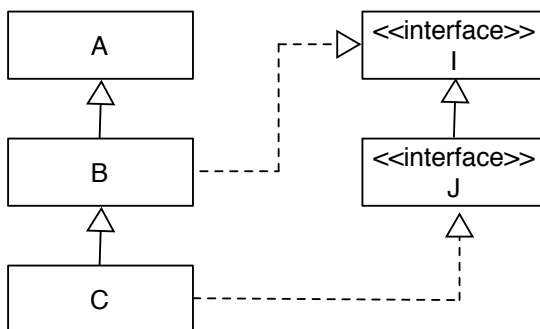


Lösningförslag

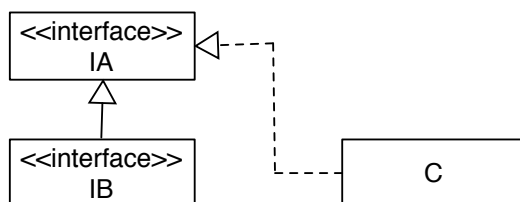
Polymorfism, skuggning och överlagring av metoder

U 1. Klassdiagram:



B är subtyp till A och I. J är subtyp till I. C är subtyp till J och B och därför även till A och I.

U 2. Klassdiagram:



1 är korrekt ty C implementerar IA. 2 är fel, C implementerar inte IB. 3 är korrekt ty IB är subtyp till IA. 4 är inte korrekt ty IA är inte subtyp till IB. 5 är korrekt, deklarerad typ för c är C och högerledet är ett objekt av typ C. 6 är inte korrekt, högerledet är av typ IA som inte är subtyp till C. 7 är fel ty högerledet är inte subtyp till C.

U 3. Metodanrop 1 och 4 är korrekta. Här råder tilldelningskompatibilitet mellan respektive aktuella och formella parametrar. 2 är fel ty den andra parametrarn uppfyller inte kravet på tilldelningskompatibilitet. 3 är fel av samma skäl. 5 är också fel. p1 har deklarerad typ Person och den andra aktuella parametrarn uppfyller därför inte kravet på tilldelningskompatibilitet.

U 4. Metodens signatur är dess namn tillsammans med antalet parametrar och deras typer.

U 5. Med skuggning (eng: override) menas att det finns metoder med samma signatur i en klass och i någon av dess superklasser eller subklasser. Exempel: metoden toString i exemplet med klasserna Person och dess subklasser från övningsuppgifter 1.

U 6. Metoder som har samma namn men olika signatur (t. ex olika antal parametrar eller olika typer på sina parametrar) sägs överlagra (eng: overload) varandra. Exempel: metoden append i klassen StringBuilder. Ett annat exempel på överlagring är när klasser har flera olika konstruktörer. Se till exempel dokumentationen för klassen ArrayList.