

EEL670 Trabalho 16

Erica Ferreira

Maio 2021

1.

Suponha a seguinte declaração Java:

```
class C1 implements I1 {  
}  
class C2 extends C1 implements I2 {  
}  
class C3 implements I1 {  
}
```

onde I1 e I2 são duas interfaces completamente distintas, nenhuma estendendo a outra, e suponha variáveis de cada tipo:

```
C1 c1;  
C2 c2;  
C3 c3;  
I1 i1;  
I2 i2;
```

Para cada uma das possíveis atribuições das cinco variáveis, diga se a linguagem Java permite, não permite, ou permite somente com *downcast*. Explique os porquês.

Resposta

c1 = c2

Permitido, *c2* faz parte de C1

c1 = c3

Permitido, uma vez que com *downcast* *c3* se torna I1 (que faz parte de C1).

c1 = i1

Permitido com *downcast*, uma vez que com C1 implemeta I1.

c1 = i2

Não permitido, não tem como torná-los compatíveis.

c2 = c1

Permitido com *downcast* uma vez que têm alguns objetos em comum.

c2 = c3

Não permitido, não tem como tornar *c2* compatível com C3.

c2 = i1

Permitido com *downcast*, uma vez que têm alguns objetos em comum.

c2 = i2

Permitido, *i2* faz parte de C2.

c3 = c1

Permitido uma vez que têm I1 em comum.

c3 = c2

Não permitido, não tem como tornar *c3* compatível com C2.

c3 = i1

Permitido com *downcast*, uma vez que com *downcast* *i1* vira C3.

c3 = i2

Não permitido, não tem como fazer serem compatíveis.

i1 = c1

Permitido, uma vez que I1 é implementada em C1.

i1 = c2

Permitida sem downcast, uma vez que C2 herda I1 de C1.

i1 = c3

Permitida sem downcast, uma vez que I1 é implementada em C3.

i1 = i2

Não permitido, interfaces completamente distintas.

i2 = c1

Não permitido, I2 não tem ligação com C1.

i2 = c2

Permitido sem downcast, uma vez que I2 é implementada em C2.

i2 = c3

Não permitido, I2 não tem ligação com C3.

i2 = i1

Não permitido, interfaces completamente distintas.