# Programmation orientée objet avancée

TD2

24/04/2021

Exercice 1 : Polymorphisme élémentaire

### Question 1

Dans le cas ou x = A, x.mA1() appelle la méthode de la classe A.

Dans le cas ou x = B, x.mA1() appelle la méthode de la classe B.

Dans le cas ou x = C, x.mA1() appelle la méthode de la classe C.

### Question 2

Dans le cas ou x = A, x.mA2() appelle la méthode de la classe A.

Dans le cas ou x = B, x.mA2() appelle la méthode de la classe A, car B ne redéfini pas la méthode mA2, mais il hérite de A.

Dans le cas ou x = C, x.mA2() appelle la méthode de la classe A, car pareil que B.

### Question 3

Erreur de compilation.

## Exercice 2 : Polymorphisme avec les méthodes d’instance et de classe

### Question 1

1er tour de boucle :  
m1 de A  
m2 de A

2eme tour de boucle :  
m1 de B  
m2 de A

### Question 2

Si on modifiait le type du tableau B[], il y aurait une erreur de compilation car B hérite de A, on ne peut pas mettre un A dans un tableau de B, il manque des éléments.

### Question 3

Si on retirait la méthode m1 de la classe A, il y aurait une erreur car si A n’a pas de m1 la méthode sera appelée dans « le vide ».

### Question 4

Si on retirait la méthode m1 de la classe B, le m1 de la classe A sera appelé.

### Question 5

La première ligne se compile correctement car t[1] est un élément de la classe B et il est donc stockable dans une variable B à l’aide d’un cast.

### Question 6

L’exécution des lignes :  
b1.m1();  
b1.m2();

Retournera :  
m1 de B  
m2 de B

### Question 7 La première ligne se compile correctement car t[0] est un élément de la classe B et il est donc stockable dans une variable B à l’aide d’un cast.

### Question 8

L’exécution de la deuxième ligne :   
b2.m1();

Ne fonctionnera pas car t[0] fait référence à une instance de A, l’affectation es donc impossible.

## Exercice 3 : Construction d’objets...

### Question 1