

POO: Contrôle continu N°1

Durée: 20 mn - Documents interdits

Questions de cours (2.5pts)

1- Expliquez la (les) différence(s) entre surdéfinition et redéfinition (0.5+0.5+0.5)

1. La redéfinition est liée à l'héritage alors que la surdéfinition peut se faire au sein d'une même classe
2. La surdéfinition impose des signatures de méthodes différentes alors que la redéfinition impose la même signature
3. La surdéfinition ne prend pas en compte le type de la valeur de retour alors que la redéfinition exige le même type de retour

2- Donnez le concept de la POO (0.5 + 0.5)

- 2-1- Manipuler des objets sans en connaître tout à fait le type: **Polymorphisme**
- 2-2- Regrouper les caractéristiques et les comportements communs à des entités: **Abstraction**

Exercice (7.5points): Identifier l'erreur en expliquant le problème :

```

public class Erreur {
    private int j = 5;
    public int h(int i){return 2*i;}
    public static void main(String[] args) {
        System.out.println(h(3)); <1> (0.5)
        E1 e1 = new E2();
        System.out.println(e1.f()*j); <2> (0.5)
        System.out.println(e1.g()); <3> (0.5)
    }
}
class E1 {
    private int i;
    public E1(int i){this.i = i;}
    public int f(){return i;}
}
class E2 extends E1 { <4> (0.5)
    public int g(){return 3*i;} <5> (0.5)
}

```

Description des problèmes:

- 1- La méthode h n'est pas static (0.5)
- 2- L'attribut j n'est pas static (0.5)
- 3- La fonction g() n'est pas accessible (0.5)
- 4- Problème de constructeur dans la classe E1 (0.5)
- 5- L'attribut i n'est pas accessible (visible) (0.5)

Correction:

- 1- rendre h static, ou créer une instance de Erreur et appeler h(int) (0.5 pour une des réponses)
- 2- rendre j static, ou créer une instance de Erreur et utiliser j même s'il est private (dans la même la classe) (0.5 pour une des réponses)
- 3- changer le type de la référence e1 à E2, ou faire une conversion explicite (cast) (0.5 pour une des réponses)
- 4- ajouter un constructeur sans arguments dans E1 ou doter E2 d'un constructeur avec un argument de type int faisant appel au constructeur de E1 avec super(int) : dans ce deuxième cas on instancie e1 comme suit e1= new E2(int) (0.5 pour la partie en noir)
- 5- rendre i protected ou utiliser un getter (la fonction f peut être utilisée comme getter) (0.5 pour une des réponses)