

Corrigé type Examen programmation orientée objet

Exercice 1: (10 points)

1. Pour chacune de ces affirmations, indiquez si elle est vraie ou fausse: (0,25 points chacune)

5. Trouvez et corriger les erreurs pour que ce programme fonctionne: (3points, 0,5 chaque bonne réponse pour la 1ère solution)

<pre>package a; class A { private int x; public A(int px){ x=px; // au lieu de px=x ; (0,5ptt) } int getX(){ return x; (0,5ptt) } }</pre>	<pre>package a; (0,5ptt) public class B extends A { private int y; public B(int px, int py){ super(px); (0,5ptt) y=py;} public int somme (){ return getX()+y //au lieu de x+y; (0,5ptt) }</pre>	<pre>package b; import a.B; class Test{ public static void main (String [] args){ B b=new B(8,7); System.out.print(b.somm e()); ; } (0,5ptt) }</pre>
---	---	--

- (a) FAUX (c) VRAI
(b) VRAI (d) VRAI

2. Pour chacune de ces affirmations, indiquez si elle est vraie ou fausse: (0,25 points chacune)

- (a) FAUX (c) VRAI
(b) FAUX (d) FAUX

3. Qu'affiche le programme suivant ? : (1,5 points)

```
class Compteur {
private int compteur = 0;
Compteur( ) { compteur += 1;}
public static void main(String[] args) {
Compteur c1 = new Compteur();
Compteur c2 = new Compteur();
Compteur c3 = c1;
Compteur c4 = c2;
System.out.println(" compteur est : " + compteur); }
}
```

(c) erreur de compilation (0,5 point) ,

Justification (1point): Attribut compteur non statique manipulé dans méthode statique. Ou bien, la variable d'instance compteur peut être appelée uniquement à travers l'objet.

4. Corrigez les constructeurs des classes A,B et ajoutez un constructeur a la

classe C: (2 points)

<pre>class A { int a; A(int a) { this.a=a; }} (0,5 ptt)</pre>	<pre>class B extends A { int b; B(int a, int b) { Super(a); (0,25 ptt) this.b=b; }(0,25 ptt) }</pre>	<pre>class C extends B { int c ; C(int a,int b,int c){ Super(a,b); (0,5 ptt) this.c=c; (0,5 ptt) }}</pre>
---	--	---

Ou 2ème solution :

<pre>package a; class A { (1 pt) protected int x; public A(int px){ x=px; // au lieu de px=x (0,5ptt) } }</pre>	<pre>package a; (0,5ptt) public class B extends A { private int y; public B(int px, int py){ super(px); (0,5ptt) y=py;} public int somme (){ return x+y; } }</pre>	<pre>package b; import a.B; class Test{ public static void main (String [] args){ B b=new B(8,7); System.out.print(b.somme()); } (0,5ptt) }</pre>	<pre>public document (String titre, String auteur) { this.titre=titre; (1 point sur le constructeur) this.auteur=auteur;} public dictionnaire(String titre, String auteur, String langue, int nbrtomes) { super(titre, auteur); this.langue=langue;(0,25 sur le constructeur) this.nbrtomes=nbrtomes;} public livre (String titre, String auteur,int nbpage, int nbcopies,booleanetat) { super(titre,auteur); (1 points sur le constructeur) this.nbrpage=nbpage; this.nbcopies=nbcopies; this.etat=etat; numLivre++;}</pre>
---	--	--	---

6. Trouver et corrigez les erreurs dans les classes ci-dessous ? (1,5 points 0,5 point chaque bonne réponse)

```
class Exemple {
static private double d;//supprimer le mot clé final
(0,5ptt) private static int i;
static int fx(int n) {i=n; return i;(0,5ptt)}
void gx(int n) {i=n; d=(double)n;
}
}public class Application
{ public static void main ( String [] args ) {
Exemple e1=new Exemple ();
int j =5;
e1.gx(j ); (0,5ptt)
e1.fx(j );
}
}
```

3

```
public String gettitre() (0,5 point){
return titre; }
public String getauteur() (0,5 point) {
return auteur; }
```

4

```
public static boolean verifieauteur(livre l1, livre l2)(0,5
point){ boolean v= (l1.auteur.equals(l2.auteur ));
return v; }
```

5

```
public int nblivres()(0,5 point){return numLivre;}
```

Exercice 2 :(10 points)

I-1

```
public class document {
String titre; (0,5 sur les attributs)
String auteur; }
public class dictionnaire extends document {
String langue;(0,25 points sur les attributs)
int nbrtomes; }
public class livre extends document{
int nbrpage;(0,5 points sur les attributs)
int nbcopies;
boolean etat;
static int numLivre=0;}
```

6

```
public void afficher () (0,5 point) {
System.out.println("le titre du livre est "+ titre+"l'auteur du livre
est "+auteur+
"le nombre de copies est "+nbcopies+ "l'état du livre est
"+etat+"le numéro du livre est"+numLivre);
}
```

7

```
public void setauteur(String Nouveauteur)(1 point) {
auteur=Nouveauteur;}
```

```
public class adhérent {
    String nom;(0,25 points )
    String prénom
    int cpt=0;
    public adhérent (String nom, String prénom) {
        this.nom=nom;(0,25 points)
        this.prénom=prénom;}}

```

2

```
public void emprunterlivre(livre l) (0,5 point){
    if ((l.nbcopies>0)&&(l.etat==true)) {
        l.etat=false;
        l.nbcopies--;
        cpt++;
    }
}
public void rendrelivre(livre l)(0,5 point) {
    l.etat=true;
    l.nbcopies++;
}

```

3

```
public int nblivreemprunté()(0,5 point){
    return cpt;
}

```

```
public class Testbibliothèque (1point)
(0,5 points sur la création des objets et 0,5 points sur l'appel
des méthodes{
    public static void main(String[] args) {
        adhérent a= new adhérent("Mohamed");
        adhérent b=new adhérent("Ali");
        livre l1=new livre("programmer en java","auteur
1",300,24,true); livre l2=new livre("programmation orientée
objet","auteur 2",10,20,true);
        livre l3=new livre("langage java","auteur
3",100,56,true); l1.afficher();
        a.emprunterlivre(l2);
        System.out.println("nbr de livres empruntés
est" +a.nblivreemprunté()); a.rendrelivre( l2);
        boolean c=livre.verifieauteur(l1, l2);
        System.out.println(c);
        setauteur("auteur 3");
        l1.getauteur();
        l1.gettitre();
        System.out.println(nblivres());
        system.out.println(a.nblivreemprunté());
    }
}

```

III