Nom et Prénom : ..... **Groupe** ..... **numListe**.....

Université de Jijel - Faculté des sciences exactes et informatique - Département d'informatique

Licence En Informatique - Programmation Orientée Objet

Examen –

---Tous les documents sont autorisés ---

Calculatrice, PC et smartphone sont interdits

Temps 1H : 30'

Exercice 01 : 06 pts *(4 x 1,5)*

1- En respectant la sémantique d'exécution, donner pour chaque morceau d'algorithme qui suit le code Java correspondant.

Fonction isEgaux (x , y : entier) : booléen
Début
isEgaux ← faux
Si (x = y) **alors**
isEgaux ← vrai
finSi
Fin
	boolean isEgaux(int x, int y) {
	if(x== y) {
	return true;
	}
	return false;
	}
	1,5pt

Procédure produire ( )
var
x : entier
Début
x ← 5
tant que (x > 0) **faire**
écrire ("la valeur de x est :", x)
x← x -2
finTq
Fin
	void produire ( ){
	int x =5 ;
	while (x>0) {
	System.out.println("la valeur de x est :" +x) ;
	x-=2 ;
	}
	}
	1,5pt

2- Que donne l'exécution des instructions suivantes

`int a = 1;`
`switch(a){`
`case 1 : System.out.println("1");`
`case 2 : System.out.println("2");break;`
`case 3 : System.out.println("3");`
`default: System.out.println("0");`
`}`
	1 **0,5pt**
	2 **1pt**

`int x = 5 ;`
`while (x>0) {`
`x-- ;`
`if(x==4) {`
`System.out.println(x);`
`continue;`
`}`
`if (x==3) {`
`System.out.println("continuer");`
`break;`
`}`
`System.out.println("passer");`
`}`
	4 **0,25**
	continuer **1,25**

Nom et Prénom : ..... Groupe ..... numListe.....

Exercice 02 : **07** pts (**2-3-2**)

Considérons les classes suivantes :

```
package un.dz;
public class A {
protected int y;
int z;
public int t ;
}
```

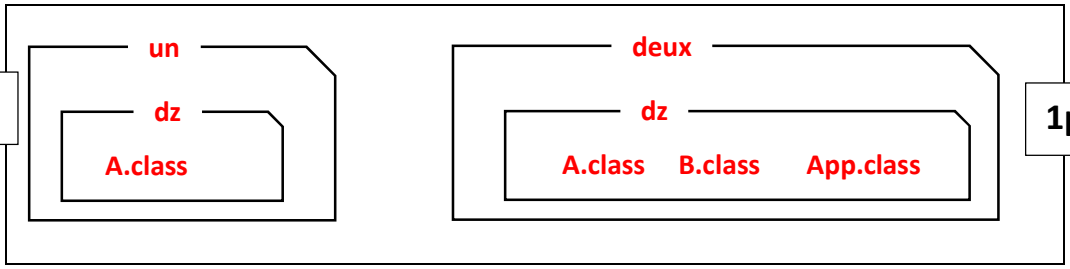
```
package deux.dz;
import un.dz.A;
public class B extends A {
  protected int b;
  int c;
}
```

```
package deux.dz;
public class C extends App {
   protected int p;
   int q;
}
```

```
package deux.dz;
public class App {
 protected int f;
 int m;
}
```

1- Donner où seront créés les fichiers **A.class** , **B.class**, **C.class** et **App.class** Après la compilation

1pt



1pt

2- La classe **App** contient la méthode **main** suivante :

2 réponses correctes = 0.75 pt

```
package deux.dz;
public class App {
protected int f;
int m
public static void main(String[] args){
```

A objA = new A();	☐
B objB = new B();	☐
C objC = new C();	☐
objA.y = 5;	☐
objA.z = 5;	☐
objB.y = 5;	☐
objB.z = 5;	☐
objB.b = 5;	☐
objB.c = 5;	☐
objC.q = 5;	☐
objC.p = 5;	☐
objC.f = 5;	☐
objC.m = 5;	☐
objA.z = 5 ;	☐
objA.y = 5;	☐
objA.t = 5;	☐

```
}}
```

Nom et Prénom : ..... Groupe ..... numListe.....

3- Considérons les classes suivantes

```
public class B {
  int x = 2;
  public String toString(){
   return x+"" ;
}
}
```

```
public class A {
  int x;
  static void incrementer(int x) {
   this.x =25;
   x++ ;
  }
  static void incrementer(B b){
     b.x++ ;
  }
}
```

```
public class Application {
public static void main(String[]){
  B b = new B();
  int a = 3;
  A.incrementer(a);
  A.incrementer(b);
  System.out.println(a);
  System.out.println(b);
}}
```

Mentionner les résultats ici

3 **1pt**

3 **1pt**

Exercice 03

```
public interface Geometrie  {
   double ANGLE = Math.PI/4;
   double surface();
   double perimetre();
}
```

0.5pt
0.25pt
0.25pt

Bonus : Constante en majuscule + 0.25pt

```java
public class Cercle extends Point implements Geometrie{

private double rayon;

public Cercle(Point p, double rayon) {
      super(p.x, p.y);
      this.rayon = rayon;
}

@Override
public double surface() {

      return Math.PI*rayon*rayon;
}
@Override
public double perimetre() {

      return 2*Math.PI*rayon;
}


public double getRayon() {

      return rayon;
}


public void setRayon(double rayon) {

      this.rayon = rayon;
}


public String toString () {

      return "Cercle de centre"+getP()
       + "est de rayon :"+rayon;
}


}
```

Bonus + 0.25pt

0.5pt

0.25pt

0.25pt

Bonus + 0.25pt

Bonus + 0.5pt

```java
public class Huit implements Geometrie{
    private Cercle haut;
    private Cercle bas;
    private Point p;

     public Huit(Point p) {
          this.p = p;
          this.haut = new Cercle(new Point(p.x,p.y+1),1);
          this.bas = new Cercle(new Point(p.x,p.y-1),1);
     }

     @Override
     public double surface() {

          return haut.surface()+bas.surface();
     }

     @Override
     public double perimetre() {

          return haut.perimetre()+bas.perimetre();
     }

public Point getP() {

     return p;
}

public void setP(Point rayon) {

     this.p = p;
}

public Cercle getHaut() {

     return haut;
}

public void setHaut(Cercle haut) {

     this.haut = haut;
}

public Cercle getBas() {

     return bas;
}

public void setBas(Cercle bas) {

     this.bas = bas;
}
```

0.25pt

Réponses correcte même si vous ne précisez pas **private**

0.5pt

0.5pt

0.5pt

Bonus 0.25pt

0.25pt

0.25pt

```
}
```

3 - Qu'affiche le programme suivant ?

Très très simple

```
public static void main(String[] args) {
Point p1 = new Point(1,1); Point p2 = p1;    // ici p2 pointe vers le même objet que p1 → p2=p1
Point p3 = p2; // ici p3 pointe vers le même objet que p2 → p3=p2
  Donc  p1=p2=p3  → p1 ,p2 et p3 pointent vers le même objet
p2.y = 4; // → p1.y = 4 et p3.y =4
p3.x =2; // → p1.x = 2 et p2.y =2
t = charger (p1,p2,p3); → la méthodes elle crée des points avec (pi.x + pi.y , 0)→ pour chaque point
   → (4+2 , 0) → (6,0)
for (Point r : t){
                System.out.println(r);
        }
    }
```

(6.0,0.0)
(6.0,0.0)
(6.0,0.0)

1pt
1pt
0.75pt