

**Calculatrices autorisées.**

**Cours autorisé.**

*Les exercices sont indépendants. Les temps donnés sont indicatifs.*

### 1–Questions de cours (30 minutes)

Répondez aux questions suivantes :

1. Donnez un exemple de classe et commentez le.
2. Donnez un exemple d'héritage.
3. Donnez un exemple de coercion de type ("cast").
4. Donnez un exemple d'utilisation de la liaison dynamique.
5. Donnez un exemple de type primitif.

### 2–Points, Droites (60 minutes)

On vous propose de créer 2 classes permettant de modéliser des points et des droites.

```
public class Point {
    public double x;
    public double y;

    public Point(double x,double y){
        this.x = x;
        /* A completer */
    }

    public double distance(Point p){
        double dX = p.x - this.x;
        double dY = /* A completer */
        return Math.sqrt(dX*dX+dY*dY);
    }

    public void translater(double dX, double dY){
        /* A completer */
    }

    public Point milieu(Point p){
        /* A completer */
    }

    public boolean equals(Point p){
        /* A completer */
    }

    public String toString(){
        return " ( x = " + x +
            " , y = " + y + " ) ";
    }
}

public class Droite {
    public Point p1 = null;
    public Point p2 = null;

    public Droite(){
        /* A completer */
    }

    /*
     * A completer
     */
}
```

**Question 1** Complétez le corps du constructeur de la classe Point. Complétez le corps de la méthode double distance(Point p).

**Question 2** Complétez le corps de la méthode `void translater(double deltaX, double deltaY)`.

**Question 3** Donnez un exemple d'appel au constructeur, à la méthode `translater`, ainsi qu'à la méthode `toString`

**Question 4** Complétez le corps de la méthode `Point milieu(Point p)`. Cette méthode doit créer et retourner un nouveau point situé au milieu des `Point this` et `p`.

**Question 5** Complétez la méthode `equals`.

**Question 6** Complétez la classe `Droite` : un droite sera représentée par une paire de points `p1` , `p2`.

Vous devrez créer des méthodes équivalentes à celles de la classe `Point` et les compléter. Vous expliquerez vos choix en ajoutant des commentaires à vos méthodes, en particulier celles de la méthode `equals` et de la méthode `milieu` dont vous définirez le comportement.

### 3-QCM (30 minutes)

**Question 1** En utilisant un mot de 8 bits, que vaut le codage *hexadécimal* de la valeur décimale  $-21$  ?

- a. FFEA
- b. E9
- c. EB
- d. EA

*Rappel : le complémentaire d'un nombre binaire se calcule en inversant les bits puis en ajoutant 1*

**Question 2** Pour quelles familles d'entiers cette fonction renvoie-t-elle toujours `true` ?

```
static boolean f(int n) {
    while((n != 1)&&(n % 2 == 0)){
        n = n / 2;
    }
    return (n == 1);
}
```

- a. les entiers  $n$  divisibles par 2.
- b. les entiers strictement positifs  $n$  divisibles par 2.
- c. les entiers  $n$  dont la valeur absolue est une puissance de 2.
- d. les entiers strictement positifs  $n$  qui sont des puissances de 2.

**Question 3** Quelle(s) instruction(s) possible(s) pour compléter ce code ?

```
LinkedList list = new LinkedList();
int a = 7;
/* A completer */
```

- a. `list.add(a);`
- b. `list.add(new Integer(a));`
- c. `list.add((Object)a);`
- d. `list.add((Object)new Integer(a));`

**Question 4** On utilise un code en virgule fixe sur 16 bits (8 bits / 8bits). Quelle est la valeur codée par 1A33

- a. environ 26,50
- b. environ 26,20
- c. environ 27,20
- d. environ 27,50

**Question 5** Comment réaliser une multiplication d'un entier positif  $x$  par  $320_d$  sans utiliser l'opérateur de multiplication ? (rappel opérateur de décalage des bits à gauche est donné par " $\ll$ "). Exemple :  $x = x \ll 3$  est équivalent à  $x = x * 8$ )

- a.  $x = x \ll 8 + x \ll 6$
- b.  $x = x \ll 320$
- c.  $x = x \ll 7 + x \ll 5$
- d.  $x = x \ll x$

**Question 6** Quelle relation entre la méthode `p1.equals(p2)` et l'instruction `p1 == p2` est toujours vraie ?

- a.  $p1 == p2 \implies p1.equals(p2)$
- b.  $p1.equals(p2) \iff p1 == p2$
- c.  $p1.equals(p2) \implies p1 == p2$
- d. elles sont totalement indépendantes