

NFA032 Programmation objet avec Java

Barème détaillé du sujet 1

Première session 2013

Exercice 1 (4 points)

question 1 (2 points)

- un point pour la structure de classe et les variables (représentation correcte des données, y compris le numéro NIRPP qui ne doit pas être un int (ça peut être un String, deux int, un tableau d'int, etc).
- un point pour les deux méthodes. La seconde doit renvoyer un boolean.

Note : le sujet ne demande pas explicitement de constructeur. Ne pas pénaliser s'il n'y en a pas.

question 2 (2 points)

- 1,5 points pour la classe NIRPP.
- 0,5 point pour la classe Personne adaptée.

Note : le sujet ne demande pas explicitement de constructeur. Ne pas pénaliser s'il n'y en a pas.

Exercice 2 (3 points)

- représentation des objets et tableaux :
 - bon nombre d'objets et tableau (deux Point, un Segment, un tableau de Point et un Polyedre : 0,75 points.
 - les objets et tableau ont le bon contenu : 0,75 points.
- représentation des références : 1,5 points.

Ne pas pénaliser si la référence sur le tableau n'est pas bonne (erreur dans l'énoncé). Ne pas pénaliser non plus si l'élève a corrigé l'erreur d'énoncé. La distinction Pile-Tas n'est pas demandée.

Exercice 3 (3 points)

Un point pour chaque trace d'exécution.

```
> java ExoExc_1
7
5
Fin normale p3
Fin normale p2
```

```
Fin normale p1
Fin programme
```

```
> java ExoExc_1
0
1
Recuperation p1
Fin programme
```

```
> java ExoExc_1
5
1
Fin normale p3
Fin normale p2
Fin normale p1
Fin programme
```

Exercice 4 (4 points)

question 1 (0,75 points)

0,75 points.

```
public class ElementListe {
    private char valeur;
    private ElementListe suivant;

    public ElementListe(char valeur , ElementListe suivant) {
        this.valeur = valeur;
        this.suivant = suivant;
    }

    public ElementListe(char v) {
        this.valeur = v;
        this.suivant = null;
    }

    public char getValeur() {
        return valeur;
    }

    public void setValeur(char valeur) {
        this.valeur = valeur;
    }

    public ElementListe getSuivant() {
        return suivant;
    }
}
```

```
    public void setSuivant(ElementListe suivant) {  
        this.suivant = suivant;  
    }  
}
```

question 2 (0,75 points)

Mettre char partout où il y a int, sauf le résultat de longueur qui reste int.
0,75 points.

```
public void ajouterAuDebut(char v) {  
    ElementListe ancienPremier= premier;  
    premier= new ElementListe(v, ancienPremier);  
}  
public void ajouterALaFin(char v) {...}  
public boolean contient(char v) { ...}  
public void retirerPremiereOccurrence(char v) {...}
```

question 3 (1,5 points)

1,5 points.

```
public void elimineRepete(){  
    ElementListe elem = premier;  
    while (elem != null){  
        while (elem.getSuivant()!=null &&  
            elem.getSuivant().getValeur()==elem.getValeur()){  
            elem.setSuivant(elem.getSuivant().getSuivant());  
        }  
        elem = elem.getSuivant();  
    }  
}
```

question 4 (1 point)

1 point.

```
public int compteRepete(){  
    ElementListe elem = premier;  
    char precedent = '_';  
    int res = 0;  
    if (elem!=null){  
        precedent=elem.getValeur();  
        elem = elem.getSuivant();  
    }  
    while (elem != null){
```

```
        if (elem.getValeur()==precedent)
            res++;

        precedent=elem.getValeur();
        elem = elem.getSuivant();
    }
    return res;
}
```

Exercice 5 (6 points)

question 1 (3 points)

```
public class PostMessage extends Post{
    private String message;
    public PostMessage(String auteur , String text){
        super(auteur);
        message = text;
    }
    public String getText(){
        return message;
    }
    public void affiche(){
        System.out.println(getText());
        super.affiche();
    }
}

public class PhotoPost extends Post{
    private String nomfichier; // fichier de l'image
    private String legende;    // legende de l'image
    public PhotoPost(String auteur , String fich , String leg){
        super(auteur);
        this.nomfichier = fich;
        this.legende = leg;
    }
    public String getImageFile(){
        return nomfichier;
    }

    public String getLegende(){
        return legende;
    }

    public void affiche(){
        System.out.println(getLegende());
        super.affiche();
    }
}
```

question 2 (1,5 point)

1,5 points.

```
public class FilActualites {
    private ArrayList<Post> posts;

    // Construit un fil vide
    public FilActualites(){
        posts = new ArrayList<Post>();
    }
    public void ajouterPost(Post post){
        posts.add(post);
    }
    public void affiche() {
        for(Post p : posts) {
            p.affiche();
            System.out.println();
        }
    }
    public int nombrePostAuteur(String a){
        int s = 0;
        for(Post p : posts) {
            if (p.getAuteur()==a)
                s++;
        }
        return s;
    }
    public int popularite(){
        int s = 0;
        for(Post p : posts) {
            s = s + p.getLikes();
        }
        return s;
    }
}
```

question 3 (1,5 point)

1,5 points.

```
public static void main(String[] args) {
    FilActualites fa = new FilActualites();
    Post p1 = new Post("Geek93");
    PhotoPost p2 = new PhotoPost("Geek93","photo1", "Foto_de_grupo");
    PostMessage p3 = new PhotoPost("Nitou","Le_temps_est_pourri");
    p2.Aime(); p2.Aime();
    p3.Aime();
    fa.ajouterPost(p1);
    fa.ajouterPost(p2);
    fa.ajouterPost(p3);
}
```

```
fa.affiche();  
System.out.println("Nombre_de_posts_de_Geek93:_ " + fa.nombrePostAuteur("Geek93"));  
}  
}
```
