



Nom :
Prénom :
Numéro d'étudiant :
Formation :

Correction contrôle 2 - Classes, héritage, ArrayList

Question 1. Savoir compléter une classe déjà définie. Etudiez le listing de la classe Jeu (listing 1).

Listing 1 – Jeu.java

```
package cc_2018_2_Jeux;

public class Jeu {
    private String nom = "inconnu";
    private int dureeParTourDeJoueur; // en minutes
    private int nbJoueurs; // en minutes
    private int ageMin; // en années

    public Jeu() {}
    public Jeu(String nom, int dureeParTourDeJoueur, int nbJoueurs, int ageMin) {
        this.setNom(nom);
        this.setDureeParTourDeJoueur(dureeParTourDeJoueur);
        this.setNbJoueurs(nbJoueurs);
        this.setAgeMin(ageMin); }

    public String getNom() {return nom;}
    public void setNom(String nom) {this.nom = nom;}

    public int getDureeParTourDeJoueur() {return dureeParTourDeJoueur;}
    public void setDureeParTourDeJoueur(int dureeParTourDeJoueur) {
        if (dureeParTourDeJoueur>0)
            this.dureeParTourDeJoueur = dureeParTourDeJoueur;}

    public int getNbJoueurs() {return nbJoueurs;}
    public void setNbJoueurs(int nbJoueurs) {
        if (nbJoueurs>0) this.nbJoueurs = nbJoueurs;}

    public int getAgeMin() {return ageMin;}
    public void setAgeMin(int ageMin) {
        if (ageMin>0) this.ageMin = ageMin;}
}
```

Ecrire une méthode `duree` permettant de calculer la durée du jeu en minutes. Cette durée se calcule comme le produit du nombre de joueurs par la durée par tour de joueur.

Réponse à la question 1 (1pt) :

```
public int duree() { \\ 0,25pt
    return \\0,25pt
        this.dureeParTourDeJoueur*this.nbJoueurs; \\0,5pt
}
```

Question 2. Savoir écrire le code d'une sous-classe

a- Ecrire l'entête et les attributs d'une classe représentant un *JeuEducatif*, qui spécialise la classe *Jeu*. Un jeu éducatif est un jeu décrit de plus par le niveau visé qui peut être : maternelle, primaire, collège, lycée, université. Il peut être ou non en ligne. La classe doit également comporter un attribut `dureeMoyconnexion` indiquant la durée moyenne en minutes de connexion pour un jeu en ligne. Sa valeur est partagée, elle est la même pour tous les jeux éducatifs (relative à la classe et non pas à une instance) et elle est initialisée à 5 par défaut.

Réponse à la question 2.a (3pt) :

```
public enum Niveau {maternelle, primaire, college, lycee, universite;} //1pt

public class JeuEducatif extends Jeu { \\0,5pt
    private Niveau niveau; \\0,5pt
    private boolean enLigne; \\0,5pt
    private static int dureeMoyconnexion = 5; \\0,5pt
}
```

b- Ecrire pour la classe *JeuEducatif* un constructeur prenant tous les paramètres nécessaires pour initialiser tous les attributs d'instance (donc pas l'attribut dont la valeur est partagée).

Réponse à la question 2.b (3pt) :

```
public JeuEducatif(String nom, int dureeParTourDeJoueur, int nbJoueurs, int ageMin, Niveau niveau,
    boolean enLigne) { \\1pt
    super(nom, dureeParTourDeJoueur, nbJoueurs, ageMin); \\1pt
    this.niveau = niveau; \\0,5pt
    this.enLigne = enLigne; \\0,5pt
}
```

c- Redéfinir (en la spécialisant) dans la classe *JeuEducatif* la méthode `duree`. La durée d'un jeu éducatif se calcule comme la durée d'un jeu normal, à laquelle on ajoute, lorsque le jeu est en ligne, la durée moyenne de connexion.

Réponse à la question 2.c (2,5pt) :

```
public int duree() { \\0,25pt
    int res = super.duree(); \\1pt
    if (enLigne) \\0,5pt
        res += JeuEducatif.dureeMoyconnexion; \\0,5pt
    return res; \\0,25pt
}
```

Question 3. Savoir manipuler une liste d'objets

a- Ecrire (1) l'entête, (2) les attributs d'une classe `ProgrammeDeJeu` avec les informations suivantes. Un programme de jeux s'adresse à des joueurs qui ont un certain âge minimum et se compose d'une succession (une liste) de jeux. La durée cumulée de tous les jeux d'un programme doit être inférieure à 180 minutes.

Réponse à la question 3.a (2,25pt) :

```
public class ProgrammeDeJeu { //0,25
    private int ageMin; //0,5
    private ArrayList<Jeu> listeJeu = new ArrayList<>(); //1,5pt
    public final static int dureeMax = 180; // non demande specifiquement BONUS de 0,5pt
}
```

b- Ecrire, pour la classe `ProgrammeDeJeu`, une méthode `dureeTotale` calculant et retournant la durée totale du programme de jeux.

Réponse à la question 3.b (3,25pt) :

```
public int dureeTotale() { //0,5pt
    int res = 0; //0,5pt
    for (Jeu j : this.listeJeu) //1pt
        res += j.duree(); //0,75pt
    return res; //0,5pt
}
```

c- Ecrire, pour la classe `ProgrammeDeJeu`, une méthode permettant d'ajouter, à la fin de la liste des jeux, un jeu qui n'y apparaît pas déjà, et à condition que l'ajout de ce jeu ne fasse pas dépasser la durée totale de 180 minutes et enfin à condition que ce jeu s'adresse à des personnes dont l'âge minimum est supérieur ou égal à l'âge minimum prévu pour le programme. En cas de problème, un message d'erreur est affiché.

Réponse à la question 3.c (4pt) :

```
public void ajoute (Jeu j) { //0,5pt
    if (this.listeJeu.contains(j)) //0,5pt
        System.out.println("jeu "+j+" deja inclus"); //0,25pt
    else if (j.getAgeMin() < this.ageMin) //0,5pt
        System.out.println("age minimum inferieur a celui du programme"); //0,25pt
    else if (j.duree()+this.dureeTotale() > dureeMax) //0,75pt
        System.out.println("la duree totale va depasser 180 mn"); //0,25pt
    else this.listeJeu.add(j); //1pt
}
```

d- Ecrire, pour la classe `ProgrammeDeJeux`, une méthode `JeuxPourNbJoueursEgalA` prenant comme paramètre un nombre de joueurs et retournant la liste des jeux admettant ce nombre de joueurs.

Réponse à la question 3.d (3pt) :

```
public ArrayList<Jeu> jeuxPourNbJoueursEgalA(int nb){ // 1pt
    ArrayList<Jeu> res = new ArrayList<>(); //0,5pt
    for (Jeu j : this.listeJeux) //0,5pt
        if (j.getNbJoueurs()==nb) //0,25pt
            res.add(j); //0,5pt
    return res; //0,25pt
}
```