

Code apogee : Nom : Prénom :	Note:
--	-------

Examen Java SMI S6 Session Normale 2022

Pour attribuer les budgets annuels aux différents laboratoires, le conseil de l'université a proposé un processus dans lequel la répartition du budget dépend du référencement des travaux de recherche des chercheurs qui sont membres du laboratoire. Pour cela, ils ont décidé de s'appuyer sur un indice appelé H-index qui est un indice ayant pour but de quantifier la productivité scientifique et l'impact d'un scientifique en fonction du niveau des citations de ses publications. Cet indice est consultable gratuitement sur « Scopus », qui est une base de données célèbre pour les articles de recherches. Par conséquent, une catégorie (A, B, C ou D) est attribuée à chaque laboratoire selon la somme globale des H-index de ses chercheurs. Un laboratoire de catégorie D profite d'un petit budget, celui de catégorie C profite d'un budget moyen, un laboratoire de catégorie B profite d'un budget plus grand et finalement un laboratoire de catégorie A profite d'un budget encore plus grand.

Je rappelle qu'une université se compose de plusieurs laboratoires et finalement chaque laboratoire se compose de plusieurs chercheurs (pour simplifier, nous avons ignoré les facultés et les départements)

```
// nous supposons avoir tous les imports nécessaires
public class Chercheur {
    private String id;
    private String nom;
    private String prenom;
    private int H_index;

    public Chercheur(String id, String nom, String prenom, int H_index) {
        this.id = .....
        this.nom = .....
        this.prenom = .....
        this.H_index = .....
    }
    //Nous supposons avoir les getters et setters equals() et hashCode() et toString()
}
```

```
Nous supposons disposer des classes exceptions ChercheurDejaExistant et ChercheurNonExistant
//permet le tri du plus grand au plus petit
public class CompH_Index implements Comparator{
    @Override
    public int compare(Object o1, Object o2) {
        .....
    }
}
```

```

public class Laboratoire {
    private String nom;
    private ArrayList<Chercheur> chercheurs;
    private int lab_Hindex;
    private char categorie;

    /* cette methode retourne la catégorie du labo en accord avec la valeur du lab_Hindex selon les
    intervalles suivant : 0=<D<40 ; 40=<C<60 ; 60=<B<80 ; 80=<A
    On suppose qu'elle est fournie */

    private char mettreAJourCategorie(){ }

    public Laboratoire(String nom) {
        this.nom = .....
        this.chercheurs = .....
        this.lab_Hindex=.....
        this.categorie=.....
    }

    public Laboratoire(String nom, ArrayList<Chercheur> chercheurs) {
        this.nom = .....
        this.chercheurs = .....
        this.categorie=.....
    }

    //on suppose que nous avons les getters et setters en cas de besoin

    public void ajouterChercheur(Chercheur chercheur)throws ChercheurDejaExistant{
        .....
    }
    public void supprimerChercheur(Chercheur chercheur)throws ChercheurNonExistant{
        .....
    }
    //on suppose que nous avons les getters et setters en cas de besoin

    public void trierChercheurs(){
        .....
    }
}

```

```
@Override  
public String toString() {  
    .....  
}  
public void afficherChercheurs(){  
    .....  
}  
}
```

```
//On suppose que le comparateur des h-index des labos est fourni  
public class CompLabH_Index implements Comparator{
```

```
public class Université {  
    private String nom;  
    private ArrayList<Laboratoire> laboratoires;  
  
    public Université(String nom) {  
        .....  
    }  
    public void ajouterLaboratoire(String nom){  
        .....  
    }  
    void ajouterLaboratoire(Laboratoire lab) {  
        .....  
    }  
    public void ajouterChercheur(Laboratoire laboratoire, Chercheur chercheur)  
    .....  
    }  
    public void supprimerChercheur(Laboratoire laboratoire, Chercheur chercheur)  
    .....  
    }  
    public void trierLaboratoires(){  
        .....  
    }  
    public void afficherLaboratoires(){  
        .....  
    }  
}
```

```

/* Terminer le code dans main() permettant d'avoir l'affichage suivant :
Laboratoire{nom=LTIM Lab_H_Index =17, categorie=D}
1 : Chercheur{nom = Tahiri, prenom = Taha, H_index=9}
2 : Chercheur{nom = Alaoui, prenom = Ali, H_index=5}
3 : Chercheur{nom = Omari, prenom = Omar, H_index=3}
Laboratoire{nom=LAMS Lab_H_Index =96, categorie=A}
1 : Chercheur{nom = Kadiri, prenom = Hossam, H_index=80}
2 : Chercheur{nom = Yousfi, prenom = Hassan, H_index=16}
affichage laboratoires
Laboratoire{nom=LAMS Lab_H_Index =96, categorie=A}
Laboratoire{nom=LTIM Lab_H_Index =17, categorie=D}
*/
public class Main {
    public static void main(String[] args) {
        Université univ=new Université("Hassan II");
        Laboratoire lab1=new Laboratoire("LTIM");
        Laboratoire lab2=new Laboratoire("LAMS");
        univ.ajouterLaboratoire(lab1);
        univ.ajouterLaboratoire(lab2);
        Chercheur ch1=new Chercheur("1", "Alaoui", "Ali", 5);
        Chercheur ch2=new Chercheur("2", "Omari", "Omar", 3);
        Chercheur ch3=new Chercheur("3", "Tahiri", "Taha", 9);
        Chercheur ch4=new Chercheur("4", "Yousfi", "Hassan", 16);
        Chercheur ch5=new Chercheur("5", "Kadiri", "Hossam", 80);

.....
.....
.....
    }
}

```