Université Bordeaux I
Examen

Master Bioinformatique - 12 avril 2017.
Module Algorithmique et Programmation.

Programmation Java

Exercice 1

Comment s'appelle la méthode qui permet de créer une instance de classe ? Ecrire un exemple pour la classe `Element` qui nécessite un paramètre de type `int`. (4pts)

Exercice 2

Pour quelle raison la méthode `main` est-elle `static` ? (2pts)

Exercice 3

Corrigez et complétez le code de la classe ci-dessous afin que le code de cette classe compile sans erreur. (4pts)

```
class MaClasse{
  private String nom;
  static int charge;
  public MaClasse (String nom){
    this->nom=nom;
  }
  public getNom(){
    return nom;
  }
  public affiche(){
    System.out.println(''mon nom est'' +nom);
}

class Suite extends MaClasse{
  private int cpt;
  public Suite(){
    super(''defaut'');
    cpt=0;
    }
  public Suite(String ident){
    super(nom);
    this->cpt=0;
    }
  public Suite(String ident){
    super(nom);
    this->cpt=0;
  }

  public int getCpt(){
    return cpt;
    }
  public add(){
    cpt++;
  }
  public void affiche()
  {
    System.out.println(''mon nom est''+nom);
    System.out.println(''mon compteur est a '' + cpt);
  }
}
class Main {
  public static void main(String args[])
  {
    Suite item1=new Suite(2);
    Suite item2=new Suite(''toto'');
    item2.add();
    MaClasse item3=new MaClasse();
    item3.setNom(''nouveau'');
    item3.add();
    MaClasse item4 = new Suite();
    item1.affiche();
    item4.affiche();
    }
}
```

4 Problème

Vous devez réaliser un programme de gestion des résultats de micro-arrays. Vous disposez d'une liste de gènes qui sont testés sur des plaques à 128 puits. Dans chaque puit, on mesure le niveau d'expression (nombre réel) de chaque gène de la liste. On peut évidemment gérer plusieurs plaques.
Vous utiliserez les classes ci-dessous dans votre programme en y ajoutant les éléments que vous jugerez nécessaires (on considère que les fonctions usuelles `saisieEntier` et `saisieChaine` sont écrites et disponibles).

```
class Gene{
  private String nom;
  public static int effectif = 0;
  private Test expression = null;
}

class Test extends Vector{
  private int result =0;
}

class Plaque extends Vector{
  public static int nbpuits = 128;
}
```

- La classe `Gene` permet de stocker les résultats d'expression.
- La classe `Test` permet de stocker les résultats d'expression pour l'ensemble des plaques.
- La classe `Plaque` permet de stocker les résultats d'une plaque.

Vous donnerez le code permettant de répondre aux questions suivantes :

1. Permettre la saisie des résultats (2pts).
2. Classer les résultats en 3 groupes en fonction des niveaux d'expression moyens pour tous les tests : groupe 1 ceux dont le niveau moyen est inférieur ou égale à 1, groupe 2 ceux dont le niveau moyen est supérieur à 1 et inférieur ou égale à 4, groupe 3 ceux dont le niveau moyen est supérieur à 4 et afficher les gènes appartenant à chaque groupe (4pts).
3. Pour chaque groupe, donnez le gène le plus représenté (2pts). ‘
4. Vous donnerez enfin une classe principale qui contiendra la fonction `main` avec un menu permettant de réaliser ces traitements (2pts).