

**CONSERVATOIRE NATIONAL DES ARTS ET MÉTIERS
CENTRE RÉGIONAL DE NOUVELLE-AQUITAINE**

ANNÉE UNIVERSITAIRE 2020 / 2021

SUJET EXAMEN dans le cadre de la COVID-19

Code et intitulé de l'UE : NFA031-1 Programmation avec Java : notions de base

N° semestre : 1

N° session : 1

Nom de l'intervenant : Laurent FALLOT

Date : 19 février 2021

Horaire : 18 heures

Durée : 2h (+15 mn pour la prise en main de l'outil et la remise du devoir)

Nombre de pages du sujet (page de garde incluse) : 5

Barème : voir sujet

Calculatrice	Oui
Dictionnaire	Oui
Notes de cours	Oui

Pour le rendu du devoir, merci de vous référer aux consignes indiquées sur le tutoriel « participer à un examen en ligne » qui vous a été fourni et indiquées dans la rubrique d'examen sur Moodle.

1. Cours (4 points)

1.1. Comparez les avantages et inconvénients entre les tableaux classiques et les ArrayList.

1.2. Dans le bout de code qui suit

```
Terminal.ecrireStringln("Resultat = " + (2 * x + 1));
```

- a) De quoi s'agit-il ?
- b) Quels sont les opérateurs ?
- c) Quels sont les opérands de * ?
- d) Si x vaut 3, que se passe-t-il ?
- e) Si x vaut toujours 3 mais qu'on supprime les parenthèses autour de $2 \times x + 1$, que se passe-t-il ?

2. Erreurs (2 points)

2.1. Dans cette correction d'un exercice, des erreurs empêchant la compilation ont été introduites. Elles apparaissent ici soulignées comme par Eclipse. Proposez une correction pour chacune d'entre elles.

```
01. class Exo4_3_2 {
02.     public static void main(char[][] args) {
03.         int chiffre = -1.0;
04.         int res;
05.         while (chiffre < 1 || chiffre > 9) {
06.             Terminal.ecrireStringln(
07.                 "Entrez le chiffre à multiplier: ");
08.             chiffre = Terminal.lireInt();
09.             if (chiffre < 1) || (chiffre > 9) {
10.                 Terminal.ecrireString(
11.                     "Ce nombre n'est pas compris entre 1");
12.                 Terminal.ecrireStringln(
13.                     " et 9. Recommencez.");
14.             }
15.         }
16.         for (int i = 1; i < 10; i++)
17.             res = i * chiffre;
18.             Terminal.ecrireInt(i);
19.             Terminal.ecrireString(" x " + chiffre + " = ");
20.             Terminal.ecrireIntln(res);
21.         }
22.     }
23. }
```

2.2. Ces erreurs corrigées, en essayant d'exécuter le programme, on obtient le message suivant :

Erreur : la méthode principale est introuvable dans la classe septembre2016Sem2.Exo4_3_2

Comment corriger l'erreur qui produit ce message ?

3. Algorithmique (4 points)

3.1. Donner les entrées, les sorties et l'algorithme d'un programme cherchant le premier nombre premier supérieur ou égal à un entier n donné. Pour rappel, un nombre est premier que s'il n'est divisible que par 1 et lui-même.

3.2. L'algorithme suivant permet de compter le nombre de mots présents dans une String. Les mots sont délimités par des caractères autres que des lettres minuscules ou majuscules. Le traduire en Java.

Entrées

texte : un chaîne de caractères.

Sortie

nbMots : le nombre de mots présents dans texte.

Variables locales

pos : la position du caractère courant de texte à analyser .

Algorithme

```
pos = 0
nbMots = 0
tant que pos < texte.length()
    tant que pos < texte.length() et
        texte.charAt(pos) n'est pas une lettre
        ajouter 1 à pos
    fin tant que
    si pos < texte.length() alors
        ajouter 1 à nbMots
        tant que pos < texte.length() et
            texte.charAt(pos) est une lettre
            ajouter 1 à pos
        fin tant que
    fin si
fin tant que
```

fin de l'algorithme

4. Simulation (2 points)

4.1. Simulez ligne à ligne l'exécution de la méthode suivante pour n valant 3. Donner le contenu du tableau tab à la fin du programme.

```
01. public class Roll3 {
02.     public static void main(String[] args) {
03.         int[] tab = { 10, 21, 32, 43 };
04.
05.         int m1 = tab[3];
06.         int m2;
07.
08.         for (int i = 0; i < tab.length; i++) {
09.             m2 = tab[i];
10.             tab[i] = m1;
11.             m1 = m2;
12.         }
13.     }
14. }
```

5. Exercices (8 points)

L'objectif de ces exercices est d'écrire quelques méthodes pour manipuler des fractions.

Les fractions seront représentées par des tableaux de deux int. La première case, celle d'indice 0, contiendra le numérateur de la fraction, la seconde, celle d'indice 1, contiendra le dénominateur.

On suppose que vous disposez d'une méthode `int pgcd (int a, int b)` qui calcule le plus grand nombre qui divise à la fois a et b. **Il n'est pas demandé de l'écrire.**

5.1. Ecrire une méthode qui reçoit le numérateur et le dénominateur d'une fraction et retourne un tableau tel que décrit plus haut. Cette méthode devra interdire les dénominateurs négatifs ou nuls.

5.2. Ecrire une nouvelle fonction qui prend en entrée une fraction sous forme de tableau comme expliqué plus haut et retourne une nouvelle fraction représentant la même valeur mais simplifiée. Rappel, pour simplifier une fraction, il faut et suffit de diviser le numérateur et le dénominateur par le plus grand nombre les divisant tous les deux.

Si vous n'avez pas su répondre à la question précédente, vous pourrez faire comme si vous y aviez répondu.

5.3. A partir de cette question, il s'agit de manipuler des ensembles de fraction. Un ensemble de fraction sera représenté par une `ArrayList<int []>`¹. Sur cette structure de données, il n'est pas possible d'utiliser les méthodes de recherche offertes car elles ne comparent pas le contenu des tableaux, mais leur référence, d'où les questions à venir.

¹ Cette forme d'utilisation de ArrayList est tout à fait autorisée en Java.

Ecrire une méthode qui cherche si une fraction reçue en paramètre est dans un ensemble, lui aussi, reçu en paramètre. Cette méthode prendra soin de simplifier la fraction avant d'effectuer la recherche.

Comme pour la question précédente, vous pouvez supposer que vous avez répondu correctement à la question précédente même si ce n'est pas le cas.

- 5.4. Ecrire une méthode prenant en paramètre une fraction f et un ensemble E et permettant d'ajouter la forme simplifiée de f à E si elle n'est pas déjà présente.

Comme pour les questions précédentes, vous pouvez supposer que vous avez répondu correctement à la question précédente même si ce n'est pas le cas.

- 5.5. Ecrire une méthode qui calcule l'union de deux ensembles. Il ne faudra pas modifier les deux ensembles reçus en paramètre, mais en créer un troisième qui sera retourné comme résultat de la méthode.
-