

Année universitaire 2016-2017

Sujet UE NFA031: Programmation avec Java : notions de base

Examen deuxième session : XX/04/2016

Responsable : François Barthélemy

Durée : 3 heures

Consignes

Aucun document n'est autorisé.

Calculatrice non autorisée

Les téléphones mobiles et autres équipements communicants (exemple : PC, tablettes, etc) doivent être éteints et rangés dans les sacs pendant toute la durée de l'épreuve.

Sujet de 5 pages, celle-ci comprise.

Le barème est donné à titre purement indicatif. Il est susceptible de changer

→ *Vérifiez que vous disposez de la totalité des pages du sujet en début d'épreuve et signalez tout problème de reprographie le cas échéant.*

Exercice 1 : exécution de programme (3 points)

Retracez l'exécution du programme suivant en indiquant l'évolution des valeurs des variables qu'il manipule. Vous pourrez utiliser un tableau comportant une colonne pour chaque variable, en précisant sur chaque ligne du tableau le numéro de ligne de code exécutée. *Nota bene* : Vous pouvez vous limiter aux instructions qui réalisent des affectations et aux conditions de la boucle.

```
1   public static void main(String [] args){
2       int res=-1;
3       int [] tab={1,2,3,6,10};
4       boolean fin= false;
5       for (int i=0; i<tab.length; i=i+1){
6           if (!fin){
7               if (tab[i+1]-tab[i]!=1){
8                   res=i;
9                   fin=true;
10              }
11          }
12      }
13      System.out.println(res);
14  }
```

Exercice 2 : programmes et boucles (3 points)

Question 1

Ecrivez un programme qui produit au moyen d'une boucle `for` l'affichage suivant :

```
1 kilomètres a pieds ca use les souliers
2 kilomètres a pieds ca use les souliers
3 kilomètres a pieds ca use les souliers
4 kilomètres a pieds ca use les souliers
5 kilomètres a pieds ca use les souliers
6 kilomètres a pieds ca use les souliers
7 kilomètres a pieds ca use les souliers
```

Question 2

Ecrivez un programme qui produit **au moyen de deux boucles imbriquées** l'affichage suivant (qui comporte 7 lignes) :

```
9 8 7 6 5 4 3 2 1 0
9 8 7 6 5 4 3 2 1 0
9 8 7 6 5 4 3 2 1 0
9 8 7 6 5 4 3 2 1 0
9 8 7 6 5 4 3 2 1 0
9 8 7 6 5 4 3 2 1 0
9 8 7 6 5 4 3 2 1 0
```

Exercice 3 : méthodes (4 points)

Question 1

Ecrivez une méthode capable de convertir les températures de degrés Celsius en degrés Fahrenheit et inversement. Cette méthode prend en paramètre une température (nombre à virgule) et un deuxième paramètre (dont vous choisirez le type) qui précise si la température donnée par le premier paramètre est en degrés Celsius ou Fahrenheit. Cette méthode ne fait aucune lecture ni aucun affichage. Elle renvoie le résultat de la conversion.

Les formules de conversions sont les suivantes où C et F notent une même température en degrés Celsius et Fahrenheit.

$$C = \frac{5}{9} * (F - 32)$$

$$F = \frac{9}{5} * C + 32$$

Question 2

Ecrivez un programme qui saisit au clavier une température et son unité (Celsius ou Fahrenheit) et affiche la même température convertie dans l'autre unité.

Exercice 4 : programme avec tableaux (6 points)

On souhaite composer le panier d'articles correspondant à une commande sur Internet en tenant compte du prix des articles et des disponibilités en stock. Le stock du magasin est représenté par 2 tableaux : `dispoStock` contient les quantités disponibles de chaque article, et `prixStock` contient leurs prix. Pour simplifier, un article est référencé par l'indice où il apparaît dans ces deux tableaux. L'article de référence n a donc son prix dans la case n de `prixStock`, et sa quantité dans la case n de `dispoStock`. Le panier de commandes contient chacune des références à commander et la quantité souhaitée. Il sera représentée au choix, soit par deux tableaux, l'un contenant la liste de références à commander, et l'autre la quantité de chaque référence, soit par un seul tableau à deux dimensions où chaque ligne contient ces informations (référence+quantité). On vous donne une partie du code où sont déclarés les tableaux permettant de représenter le stock. On suppose que ces tableaux sont correctement initialisés. Complétez ce code en réalisant les actions suivantes :

1. Lecture d'une commande :
 - (a) lire un nombre de références à commander, puis
 - (b) lire chaque référence à commander et la quantité souhaitée. Les enregistrer dans deux tableaux ou dans un tableau à deux dimensions.
2. Vérifier et valider la commande pour chaque référence :
 - (a) pour chaque référence vérifier si elle est valide (correspond à un indice valide du stock),
 - (b) vérifier qu'elle est disponible en quantité suffisante, et dans ce cas, calculer et afficher le prix pour cette partie de la commande. Retirer ensuite cette quantité du stock.
 - (c) si la quantité est insuffisante, ou si la référence est invalide, afficher une erreur sans arrêter le programme.
3. Afficher le prix total à payer.

Code fourni :

```
public class ExoCommandeStock {
    public static void main(String[] args) {
        // Exemples d'initialisation pour le stock du magasin
        int [] dispoStock = {20,10,3, 15, 2, 17, 30};
        double [] prixStock = {60,23.50, 65, 89, 110, 22};
        // A completer
    }
}
```

Corrigé : (5 pts)

```
public class ExoCommandeStock {
    public static void main(String[] args) {
        int [] dispoStock = {20,10,3, 15, 2, 17, 30};
        double [] prixStock = {60,23.50, 65, 89, 110, 22};
        System.out.print("Nombre de références à commander? ");
        // Lecture de la commande
        int n= Terminal.lireInt();
        int [][] panier = new int[n][2];
        for(int i=0; i<panier.length; i++){
            System.out.print("Entrez une référence à commander: ");
            panier[i][0] = Terminal.lireInt();
            System.out.print("Quantité souhaitée? ");
            panier[i][1] = Terminal.lireInt();
        }
        // Validation commande, calcul et affichage
        double total=0;
        for(int i=0; i<panier.length; i++){
            int ref= panier[i][0];
            int q = panier[i][1];
            if (ref<0 || ref>= dispoStock.length)
                System.out.println(ref+ " inexistante.");
            else {
                int reste= dispoStock[ref]-q;
                if (reste<0)
                    System.out.println(ref+ ": quantité insuffisante!");
                else {
                    double p = q*prixStock[ref];
                    System.out.println(ref+ " x " +q+ " = " + p + " euros.");
                    prixStock[ref]= reste;
                    total=total+p;
                }
            }
        }
        System.out.println("Total à payer: " +total+ " euros.");
    }
}
```

Exercice 5 : méthodes et tableaux (4 points)

Question 1

`indicePremRepet(int []t)` qui renvoie l'indice où commence la première succession d'entiers identiques dans le tableau `t`. Si aucun entier du tableau ne se répète de manière successive, la fonction renvoie un nombre négatif. Ex : si `t={ 2, 1, 2, 0, 3, 3, 3, 5, 5, 5, 10, 11}`, la fonction renvoie 4.

Question 2

Ecrivez une méthode `indicePGEcart(int [] t)` qui retourne le plus grand écart (en valeur absolue) entre deux entiers consécutifs du tableau `t`.

Par exemple, si le tableau est `t={ 2, 1, 2, 0, 3, 3, 3, 5, 5, 5, 10, 11}`, la fonction renvoie 5 qui est l'écart entre 5 et 10.