

TD Séquentiel - Rappels de base

Résumé

Ces exercices ont pour but de vérifier que vous avez fixé les bases du LDA et de la programmation Java.

1 Variables et types	2
1.1 Les variables et les types	2
1.2 Java est un langage fortement typé.	3
2 Opérateurs et expressions	4
2.1 Opérateurs et expressions	4
3 L'affectation d'une valeur à une variable	5
3.1 Affectation interne en algo	5
3.2 Affectation interne en Java	6
3.3 Affectation externe en algo	8
3.4 Affectation externe en Java	8
3.5 Communication des résultats en algo	9
3.6 Communication des résultats en Java	10
4 Structure générale	11
4.1 Structure générale d'un algorithme	11
4.2 Structure générale d'un programme Java, comment le compiler et l'exécuter.	12
5 Exercices complets	15
5.1 À vous de jouer...	15



1 Variables et types

1.1 Les variables et les types

Le type des données

Comment déclarer en algorithmique les variables dont le nom est le mot en gras :

- un **montant** d'un ticket de caisse ? _____
- une **cote** sur le bulletin (pas de demi-point autorisé) ? _____
- le **titre** d'un film ? _____
- l'initiale de votre **nom** ? _____

Le type des données

Comment déclarer en Java :

- un montant d'un ticket de caisse ? _____
- une cote sur le bulletin (pas de demi-point autorisé) ? _____
- le titre d'un film ? _____
- l'initiale de votre nom ? _____

Types Java

Donnez l'équivalent Java des types vus en algorithmique.

entier _____

réel _____

chaîne _____

chaîne d'un seul caractère _____

booléen _____

Conventions de nom

Cochez les noms qui respectent les conventions (sachant que nous travaillons dans le cadre de la tva)

- ☐ double Taxe;
- ☐ final double TVA;
- ☐ double taxeValeurAjoutée;
- ☐ final double TAUXTVA;
- ☐ final double TAUX_TVA
- ☐ double t;

final

Soit le code :

```
public class Final {  
    public static void main (String[] args) {  
        final double tva = 21.5;  
        System.out.println ((1000/100)*tva);  
        tva = 19;  
    }  
}
```

qui, à la compilation, génère le message d'erreur

`Final.java:5:: cannot assign a value to final variable tva`

ce message veut dire que

- ☐ le nom tva ne respecte pas la convention de nom
- ☐ final n'est pas un mot reconnu
- ☐ on ne peut assigner une deuxième fois une valeur à une variable final
- ☐ on ne peut assigner une valeur entière à tva

Si vous n'avez pas répondu correctement à toutes les questions, au nom de variable près, révisiez ici (www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/unit_VariablesEtTypes.html)

1.2 Java est un langage fortement typé.

Le type des données

Toute donnée a un type. Il existe 3 grands types de données. Lesquels ?

— _____
— _____
— _____

Si vous n'avez pas répondu correctement à toutes les questions, révisiez ici (www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/VariablesEtTypes_learningObject3.html)

Les types primitifs numériques entiers

Donnez les types primitifs numériques entiers dans l'ordre (de celui qui prend le moins de place en mémoire à celui qui en prend le plus) :

— _____
— _____
— _____
— _____

— _____

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici
(www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/VariablesEtTypes_learningObject3.html)

Les types primitifs numériques flottants

Donnez les types primitifs numériques flottants dans l'ordre (de celui qui prend le moins de place en mémoire à celui qui en prend le plus) :

— _____
— _____

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici
(www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/VariablesEtTypes_learningObject3.html)

Le type primitif non numérique

Donnez le type primitif non numérique :

— _____

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici
(www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/VariablesEtTypes_learningObject3.html)

2 Opérateurs et expressions

2.1 Opérateurs et expressions

Opérateurs Java

Donnez l'équivalent Java des opérateurs vus en algorithmique.

* _____

/ _____ entre 2 réels

DIV _____ entre 2 entiers

MOD _____

Que vaut ?

Que vaut l'expression en algo 5 DIV 3 : _____

Que vaut ?

Que vaut l'expression en algo $5 \text{ MOD } 3$: __

Que vaut ?

Que vaut l'expression java $11/2$: __

Que vaut ?

Que vaut l'expression java $11.0/2$: ____

Que vaut ?

Que vaut l'expression java $11\%3$: __ Si vous n'avez pas répondu correctement à toutes les questions, révisez ici (www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/unit_OperateursEtExpressions.html)

Comment calculer ?

Comment calculer en Java la racine carrée d'une variable nommée nb ? : _____ Si vous n'avez pas répondu correctement à la question, révisez ici (www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/OperateursEtExpressions_learningObject10.html)

3 L'affectation d'une valeur à une variable

Cette opération est probablement l'opération la plus importante. En effet, une variable ne prend son sens réel que si elle reçoit à un moment donné une valeur. Il y a deux moyens de donner une valeur à une variable.

3.1 Affectation interne en algo

Les exemples d'affectation sont-ils corrects ?

□

```
somme, nombre1, nombre2 : entiers
nombre1 ← 3
nombre2 ← 5
somme ← nombre1 + nombre2
```

☐ `denRes, den1, den2 : réels`
`den1 ← 3.5`
`den2 ← 1`
`denRes ← den1 * den2`

☐ `cpt : entier`
`cpt ← cpt + 1`

☐ `delta, a, b, c : réels`
`a ← 4`
`b ← 3/2`
`c ← 7/4`
`delta ← b**2 - 4*a*c`

☐ `maChaine : chaine`
`maChaine ← "Bonjour"`

☐ `a, b : entier`
`test : booléen`
`a ← 4`
`b ← 3`
`test ← a = b`

☐ `somme : entier`
`somme + 1 ← 3`

☐ `somme, n : entiers`
`n ← 4`
`somme ← 3n`

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici
(www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Affectation_learningObject1.html)

3.2 Affectation interne en Java

Les exemples d'affectation sont-ils corrects ?

☐ `int somme;`
`int nombre1;`
`int nombre 2;`
`nombre1 = 3;`
`nombre2 = -8;`

```
somme = nombre1 + nombre2;
```



```
double denRes;  
double den1;  
double den2;  
den1 = 3.5;  
den2 = 1;  
denRes = den1 * den2;
```



```
int cpt;  
cpt = cpt + 1;
```



```
double delta;  
double a;  
double b;  
double c;  
a = 4;  
b = 3.0/2;  
c = 7.0/4;  
delta = b*b - 4*a*c;
```



```
String maChaine;  
maChaine = "Bonjour";
```



```
int a;  
int b;  
boolean test;  
a = 4;  
b = 3;  
test = a = b;
```



```
int somme;  
somme + 1 = 3;
```



```
int somme;  
int n;  
n = 4;  
somme = 3n;
```

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici
(www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/Affectation_learningObject2.html)

3.3 Affectation externe en algo

Les exemples d'affectation sont-ils corrects ?

- ☐ nombre1, nombre2 : entiers
demander nombre1, nombre2
- ☐ nombre1 : entier
demander nombre1
- ☐ maChaine : chaine
demander maChaine
- ☐ maChaine : chaine
demander "Entrez une chaine", maChaine
- ☐ demander nombre1 + nombre2

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici
(www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Affectation_learningObject3.html)

3.4 Affectation externe en Java

Sélection multiple

Quels morceaux de code, parmi les suivants, NE sont PAS corrects ? Pourquoi ?

- ☐ code 1

```
public class Exercice {  
    public static void main(String[] args) {  
        Scanner clavier = new Scanner(System.in);  
        int nombre1;  
        nombre1 = clavier.nextInt();  
    }  
}
```

- ☐ code 2

```
import java.util.Scanner;  
public class Exercice {  
    public static void main(String[] args) {  
        Scanner clavier = new Scanner(System.in);  
        int nombre1;  
        nombre1 = clavier.nextInt();  
    }  
}
```

- ☐ code 3

```
import java.util.Scanner;  
public class Exercice {  
    public static void main(String[] args ) {  
        Scanner clavier = new Scanner(System.in);
```



```

        double nombre1;
        nombre1 = clavier.nextDouble();
    }
}

```

❑ code 4

```

import java.util.Scanner;
public class Exercice {
    public static void main(String[] args) {
        int nombre1;
        nombre1 = clavier.nextInt();
    }
}

```

Méthodes de Scanner

Complétez les méthodes suivantes afin de permettre une lecture au clavier adéquate.

```

import java.util.Scanner;
public class ReadingExercice {
    public static void main (String[] args) {
        Scanner keyboard = new Scanner(System.in);
        int integerNb = keyboard. _____ ;
        double realNb = keyboard. _____ ;
        boolean ok = keyboard. _____ ;
        String string1 = keyboard. _____ ; // We read a word
        String string2 = keyboard. _____ ; // We read a line
        char aCharacter = keyboard. _____ ;
    }
}

```

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici (www.heb.be/esi/InitSeq/fr/../../TDSSeq/fr/html/Affectation_learningObject4.html)

3.5 Communication des résultats en algo

Les exemples de communication des résultats sont-ils corrects ?

❑

```

somme, nombre1, nombre2 : entiers
nombre1 ← 3
nombre2 ← 5
somme ← nombre1 + nombre2
afficher somme

```

❑

```

denRes, den1, den2 : réels
den1 ← 3.5

```

- ```

den2 ← 1
denRes ← den1 * den2
afficher den1, "/", den2, " ", denRes

```
- ☐
- ```

cpt : entier
afficher cpt

```
- ☐
- ```

a, b, c : réels
a ← 4
b ← 3/2
c ← 7/4
afficher b**2 - 4*a*c

```
- ☐
- ```

afficher "Bonjour"

```
- ☐
- ```

a, b : entiers
test : booléen
a ← 4
b ← 3
afficher a = b

```

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici  
[www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Affectation\\_learningObject5.html](http://www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Affectation_learningObject5.html))

### 3.6 Communication des résultats en Java

#### Sélection multiple

Quels morceaux de code, parmi les suivants, sont corrects ? Pourquoi ?

- ☐ code 1

```

import java.util.Scanner;
public class Exercice {
 public static void main(String[] args) {
 Scanner clavier = new Scanner(System.in);
 int nombre1;
 nombre1 = clavier.nextInt();
 System.out.println(nombre1);
 }
}

```

- ☐ code 2

```

import java.util.Scanner;
public class Exercice {
 public static void main(String[] args) {
 Scanner clavier = new Scanner(System.in);
 int nombre1;
 int nombre2;
 }
}

```

```

 nombre1 = clavier.nextInt();
 nombre2 = clavier.nextInt();
 System.out.println(nombre1 + nombre2);
 }
}

```

❑ code 3

```

import java.util.Scanner;
public class Exercice {
 public static void main(String[] args) {
 Scanner clavier = new Scanner(System.in);
 int nombre1;
 int nombre2;
 nombre1 = clavier.nextInt();
 nombre2 = clavier.nextInt();
 System.out.println(nombre1 + " " + nombre2);
 }
}

```

❑ code 4

```

import java.util.Scanner;
public class Exercice {
 public static void main(String[] args) {
 Scanner clavier = new Scanner(System.in);
 int nombre1;
 int nombre2;
 nombre1 = clavier.nextInt();
 nombre2 = clavier.nextInt();
 System.out.println(nombre1, nombre2);
 }
}

```

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici ([www.heb.be/esi/InitSeq/fr/../../TDSSeq/fr/html/Affectation\\_learningObject6.html](http://www.heb.be/esi/InitSeq/fr/../../TDSSeq/fr/html/Affectation_learningObject6.html))

## 4 Structure générale

### 4.1 Structure générale d'un algorithme

Pour ces exercices, nous vous demandons de comprendre des algorithmes donnés.

#### Compréhension

Que vont-ils retourner/afficher si à chaque fois les deux nombres reçus/lus au départ sont successivement 2 et 3 ?

— algorithme `exerciceA(b, a : entiers) → réel`  
   quotient : réel

```

 quotient ← a / b
 afficher quotient
fin algorithme

—
algorithme exerciceB() → entier
 a, b, c, d : entiers
 demander c, d
 a ← 2*c+5*d
 b ← 2+c*3+d
 c ← a MOD b
 retourner a DIV c
fin algorithme

—
algorithme exerciceC(x, y : réels) → réel
 x ← x*x
 x ← x*x+y*y
 x ← √x
 retourner x
fin algorithme

—
algorithme exerciceD()
 x, y : réels
 demander x, x
 x ← x MOD x + (x + 1) DIV 2
 afficher x + 3
fin algorithme

—

```

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici ([www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Structure\\_learningObject1.html](http://www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Structure_learningObject1.html))

## 4.2 Structure générale d'un programme Java, comment le compiler et l'exécuter.

### Structure d'un programme

Quelles structures générales d'un programme sont correctes (c-à-d qu'il doit compiler et respecter les conventions) parmi les suivantes ?

- ☐ code 1
 

```

public class exercice {
 // put methods here
}

```
- ☐ code 2

- ```

        public CLASS Exercice {
            // put methods here
        }

```
- ☐ code 3
- ```

 public class Exercice {
 // put methods here
 }

```
- ☐ code 4
- ```

        public class MonExercice {
            // put methods here
        }

```
- ☐ code 5
- ```

 PUBLIC CLASS EXERCICE {
 // put methods here
 }

```

### Nom d'un programme

Quels noms doivent avoir les fichiers dans lesquels sont placés les programmes suivants :

```

public class Exercice {
 // Methods
}

```

```

public class SommeChiffres {
 // Methods
}

```

```

public class sommechiffres {
 // Methods
}

```

### Compiler/exécuter

Quelle commande permet de compiler le fichier nommé `SommeChiffres.java` ?

\_\_\_\_\_

Quelle commande permet d'exécuter ce programme ?

\_\_\_\_\_

## Méthode principale

Comment s'écrit l'entête de la méthode principale (1 mot par case)?

\_\_\_\_\_ ( \_\_\_\_\_ )

## Compréhension

Que vont afficher les codes java suivants si à chaque fois les deux nombres lus au départ sont successivement 2 et 3?

```
import java.util.Scanner;
public class Exercice1 {
 public static void main(String [] args) {
 Scanner clavier = new Scanner(System.in);
 int nb1 = clavier.nextInt();
 int nb2 = clavier.nextInt();
 System.out.println(nb1 + " " + nb2);
 }
}
```

```
import java.util.Scanner;
public class Exercice2 {
 public static void main(String [] args) {
 Scanner clavier = new Scanner(System.in);
 int nb1 = clavier.nextInt();
 int nb2 = clavier.nextInt();
 int nb3 = 2*nb1 + nb2;
 System.out.println(nb3);
 }
}
```

```
import java.util.Scanner;
public class Exercice3 {
 public static void main(String [] args) {
 Scanner clavier = new Scanner(System.in);
 int nb1 = clavier.nextInt();
 int nb2 = clavier.nextInt();
 System.out.println(nb2/nb1);
 }
}
```

```
import java.util.Scanner;
public class Exercice4 {
 public static void main(String [] args) {
 Scanner clavier = new Scanner(System.in);
 int nb1 = clavier.nextInt();
 int nb2 = clavier.nextInt();
 System.out.println(nb1%nb2);
 }
}
```

```
import java.util.Scanner;
public class Exercice5 {
 public static void main(String [] args) {
 Scanner clavier = new Scanner(System.in);
 int nb1 = clavier.nextInt();
 nb1 = clavier.nextInt();
 nb1 = nb1 * nb1;
 System.out.println(Math.sqrt(nb1));
 }
}
```

Si vous n'avez pas répondu correctement à toutes les questions, révisez ici ([www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Structure\\_learningObject2.html](http://www.heb.be/esi/InitSeq/fr/../../TDSeq/fr/html/Structure_learningObject2.html))

## 5 Exercices complets

### 5.1 À vous de jouer...

#### Consignes

Il est temps de se lancer et d'écrire vos premiers algorithmes et programmes Java correspondant. Voici quelques conseils pour vous guider dans la résolution de tels problèmes :

- il convient d'abord de bien comprendre le problème posé ; assurez-vous qu'il est parfaitement spécifié ;
- résolvez le problème via quelques exemples précis ;
- mettez en évidence les variables «**données** », les variables «**résultats** » et les variables de travail ;
- n'hésitez pas à faire une ébauche de résolution en français avant d'élaborer l'algorithme définitif pseudo-codé ;
- déclarez ensuite les variables (et leur type) qui interviennent dans l'algorithme ; les noms des variables risquant de ne pas être suffisamment explicites.
- Écrivez la partie algorithmique **AVANT** de vous lancer dans la programmation en Java.

#### Exercices

Écrivez les algorithmes et codez les programmes Java correspondant qui

1. réalisent la permutation du contenu de deux variables.

2. étant donné un moment dans la journée donné par trois nombres, à savoir, heure, minute et seconde, calculent et retournent le nombre de secondes écoulées depuis minuit.
3. étant donné un temps écoulé dans la journée exprimé en secondes, calculent et affichent ce temps sous la forme de trois nombres (heure, minute et seconde).

Ex : 10000 secondes donnera 2h 46'40"

Écrivez un algorithme qui reçoit un nombre pouvant prendre des valeurs de 100 à 999 et qui retourne ce même nombre mais renversé. Le dernier chiffre devient le premier et inversement.

Par exemple : le nombre 123 devient 321 et le nombre 410 deviendra 14.

Pour plus d'exercices, révisez ici ([www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/unit\\_Exercices.html](http://www.heb.be/esi/InitSeq/fr/../../../../TDSeq/fr/html/unit_Exercices.html))