

Programmation java

Tableaux

Pierre Ortegat

EPHEC

Table of contents

1. Les tableaux
2. Exercices I
3. Le for et les tableaux
4. Exercices II

Les tableaux

Souvent il est logique de rassembler ensembles certaines données qui vont *logiquement* ensembles.

Souvent il est logique de rassembler ensembles certaines données qui vont *logiquement* ensembles.

Par exemple:

- Une liste de numéro d'articles / d'employés / ...
- Une liste d'onglets (par exemple dans le cas d'un navigateur)
- Un ensemble de mail
- etc

Créer un tableau vide

Créer une variable de type int:

```
1 int maVariable1;  
2 int maVariable2 = 42;
```

Créer un tableau de type int:

```
1 // attention, le tableau suivant n'est PAS initialisé  
  !  
2 int [] monTableau1;  
3  
4 // initialise un tableau rempli avec la valeur par  
  d faut du type  
5 int [] monTableau2 = new int [2];
```

Créer un tableau rempli

Créer une variable de type int déjà remplie:

```
1 int maVariable = 42;
```

Créer un tableau de type int rempli:

```
1 int [] monTableau = { 42, 1337 };
```

Accéder aux valeurs du tableau

Pour lire la valeur présente dans un emplacement du tableau (par exemple, le deuxième emplacement):

```
1  
2 int [] monTableau = new int [] {5, 42};  
3  
4 System.out.println(monTableau[1]);
```

Attention ! Les tableaux sont indexés à partir de 0.

Remplir le tableau

Mettre un élément dans la première case du tableau:

```
1  int [] monTableau = new int [4];  
2  
3  System.out.println(monTableau[0]);  
4  
5  monTableau[0] = 118218;  
6  
7  System.out.println(monTableau[0]);
```

Lire la taille d'un tableau

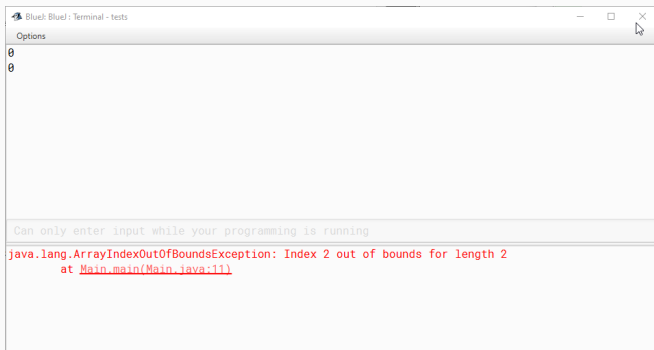
Pour récupérer la taille d'un tableau, utiliser la syntaxe: `nom du tableau.length`

```
1  int [] monTableau = new int [4];  
2  
3  System.out.println(monTableau.length);
```

IndexOutOfBounds

Si le programme tente d'accéder a une case mémoire "impossible", une erreur va se produire durant son exécution.

Si votre programme a ce problème, vérifiez que l'*index* (= la case mémoire) que vous tentez d'accéder est compris entre 0 et la taille du tableau moins un..



```
BlueJ: BlueJ : Terminal - tests
Options
0
0

Can only enter input while your programming is running
java.lang.ArrayIndexOutOfBoundsException: Index 2 out of bounds for length 2
    at Main.main(Main.java:11)
```

Exercices I

Écrire un programme qui demande à l'utilisateur dix nombre et les affichera en ordre inverse (si l'utilisateur entre 1, 2, 3, 4, 5 afficher en retour 5, 4, 3, 2, 1).

Écrire un programme qui initialise un tableau de la longueur entrée par l'utilisateur.

Ensuite remplir chacune des cases du tableau avec son indice (= son emplacement: la première case contiendra 0, la deuxième 1 et ainsi de suite).

Enfin, afficher la somme des nombre situés dans des index pairs du tableau.

Le for et les tableaux

Le for et les tableaux

Pour parcourir un tableau, un "for" permettra d'écrire du code plus facile à lire et plus concis.

Exemple:

```
1  int [] a = new int [5] ;  
2  
3  for (int i = 0 ; i < a.length ; i++) {  
4      System.out.println(a[i]);  
5  }
```


Exercices II

Écrire un programme qui instancie un tableau de la longueur entrée par l'utilisateur, le rempli avec le suite de Fibonacci en demandant l'utilisateur les deux nombres de départ et l'affiche.

Pour rappel, le suite de Fibonacci est une suite de nombre dont le suivant est la somme des deux précédents. Par exemple; ceci est une suite valide (avec 0 et 1 comme point de départ):

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Écrire un programme qui demande dix nombres à l'utilisateur et les stocke dans un tableau.

Trier ce tableau par ordre croissant et l'afficher.

Attention, ce problème a particulièrement beaucoup de solutions différentes et valides. Elles ne sont juste pas toutes aussi efficaces.

Contrainte: écrire trois versions différents marchant de façon fondamentalement différentes. Si vous manquez d'inspiration, demandez moi un indice (en cours ou via mail, p.ortegat@ephec.be).