

# MOOC Init Prog Java

## Exercices facultatifs semaine 4

### Éléments en indice (niveau 2)

Écrivez un programme appelé `elementsEnIndice` qui à partir d'un tableau `T` d'entiers construit un nouveau tableau de la façon suivante : les éléments de `T` d'indice pair seront placés dans ce nouveau tableau à l'indice donné par l'élément suivant de `T`:

- `T[0]` sera placé dans le nouveau tableau à l'indice `T[1]`,
- `T[2]` sera placé dans le nouveau tableau à l'indice `T[3]`,
- etc...

Testez votre programme avec le tableau suivant:

```
int[] tab = {4, 2, 8, 0, 7, 1};
```

Le tableau résultant devrait contenir les éléments `{8, 7, 4}`.

---

## Crible d'Ératosthène (niveau 2)

Un nombre est dit premier s'il admet exactement 2 diviseurs *distincts* (1 et lui-même). 1 n'est donc pas premier.

Le crible d'Ératosthène est une méthode de recherche des nombres premiers plus petits qu'un entier naturel  $n$  donné. Cette méthode est simple:

- On commence par supprimer tous les multiples de 2 inférieurs à  $n$ .
- L'entier 3 n'a pas été supprimé et il ne peut être multiple des entiers qui le précèdent, sinon on l'aurait supprimé; il est donc premier. Supprimons alors tous les multiples de 3 inférieurs à  $n$ .
- L'entier 5 n'a pas été supprimé, il est donc premier. Supprimons tous les multiples de 5 inférieurs à  $n$ .
- Et ainsi de suite jusqu'à  $n$ . Les valeurs n'ayant pas été supprimées sont les nombres entiers plus petits que  $n$ .

Écrivez le code qui applique cette méthode pour trouver les nombres premiers inférieurs à 100. Vous devez trouver: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97.

On utilisera un tableau de booléens:

```
boolean[] supprime = new boolean[100];
```

pour mémoriser les entiers qui ont été supprimés. N'oubliez pas d'initialiser chacun de ses éléments à `false`.

---