

Utilisez comme pseudo le nom donné sur la dernière feuille. N'oubliez pas d'inscrire vos nom et numéro avant de cacheter l'en tête de la copie.

**Remarques et recommandations préalables :**

- lorsqu'il est demandé d'écrire du code ou des fonctions, le langage utilisé est Java ;
- le correcteur n'attachera pas une grande importance à l'absence de quelques points-virgules ;
- il n'en sera pas de même, par exemple, en ce qui concerne la rigueur apportée à la déclaration des variables, la syntaxe des structures de contrôle et l'utilisation des accolades { et } ;
- la présentation (indentations) et les commentaires facilitant la compréhension de la stratégie mise en œuvre par la solution seront également appréciés ;
- les différents exercices sont indépendants.

**Exercice 1.**

---

Que donne l'exécution du programme Java suivant :

```
import fr.jussieu.script.*;
class Affectations{
    public static void main (String[ ] st) {
        int x; int y = 4; int z = 6;
        double r1, r2;
        x = y + 2; Deug.println(x);
        x = x + 4; Deug.println(x);
        x = z / y; Deug.println(x);
        r1 = z / y; Deug.println(r1);
        r1 = y; r2 = z / r1; Deug.println(r2);
    }
}
```

**Exercice 2.**

Dans cet exercice, il n'est pas demandé d'écrire un programme, mais de décrire des algorithmes dans un langage compréhensible (un programme Java est cependant une solution!).

**2.1.** On dispose de trois pièces de monnaie apparemment identiques nommées **a**, **b** et **c** dont on sait qu'au plus une est fautive (elle n'a pas le même poids que les deux autres). On dispose d'une balance permettant de comparer le poids des objets posés sur chacun des deux plateaux. Décrire un algorithme permettant de tester si une des trois pièces est fautive et si oui laquelle (pour alléger l'écriture, on pourra supposer que la valeur d'une variable  $x$  correspondant à une pièce est égale au poids de cette pièce).

**2.2.** Décrire un algorithme pour 4 pièces **a**, **b**, **c** et **d** (au plus une étant fautive) en procédant uniquement par comparaisons du poids de paires de pièces (on posera toujours 2 pièces sur chaque plateau).

*Remarque :* au cours d'une pesée, une pièce ne peut apparaître qu'une seule fois!

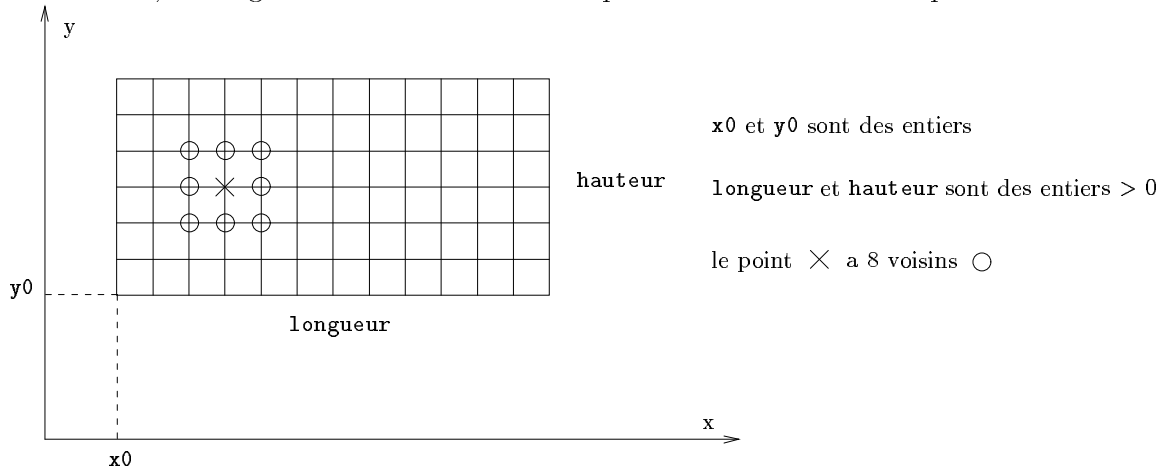
---

**Exercice 3.**

L'énoncé de l'exercice est sur la feuille jointe sur laquelle vous répondrez directement aux questions.

### Exercice 4.

Dans cet exercice, on suppose qu'on dispose d'une grille dans le plan définie par les coordonnées de son coin inférieur gauche, sa longueur et sa hauteur, toutes ces valeurs étant entières, la longueur et la hauteur étant par ailleurs strictement positives :



On se propose d'écrire un programme déterminant combien de voisins sur la grille possède un point de la grille. Pour cela, on va écrire successivement un certain nombre de fonctions simples : dans chacune de ces fonctions, la grille sera définie comme dans l'introduction de cet exercice.

- 4.1. Écrire une fonction qui, étant donné une grille et un point du plan défini par ses coordonnées, renvoie la valeur booléenne `true` si le point est sur la grille et `false` sinon.
- 4.2. Écrire une fonction qui, étant donné une grille et un point, renvoie `true` si et seulement si le point est sur la bordure de la grille.
- 4.3. Écrire une fonction qui, étant donné une grille et un point, renvoie `true` si et seulement si le point est un des coins de la grille.
- 4.4. Écrire une fonction qui, étant donné une grille et un point, renvoie le nombre de voisins du point sur la grille si ce point est sur la grille et 0 sinon.
- 4.5. Écrire un programme qui, après avoir lu la définition d'une grille et les coordonnées d'un point données par l'utilisateur, affiche le nombre de voisins sur cette grille de ce point.

---

### Exercice 5.

La médiane d'une suite finie de longueur impaire de nombres (supposés tous différents) est le nombre appartenant à cette suite qui y possède autant d'éléments qui lui sont inférieurs que d'éléments qui lui sont supérieurs.

Ainsi la médiane de  $[3, 6, 2, 1, 5, 4, 8]$  est 4.

Écrire une fonction qui, étant donné un tableau `t` d'entiers et un entier `x`, renvoie la valeur booléenne `true` si `x` est la médiane des éléments du tableau `t` et `false` sinon (en particulier si le nombre `x` n'est pas dans le tableau). On supposera que tous les éléments de `t` sont différents et qu'il contient un nombre impair d'éléments, ce qu'il n'est pas demandé de vérifier.

---

### Exercice 6.

Écrire une fonction `begaie` qui prend en argument un **tableau de caractères** et qui renvoie un nouveau tableau de caractères déduit du premier en dédoublant chacun de ses caractères (on n'utilisera pas les chaînes de caractères).

Par exemple :

t	a	b	l	e	a	u
---	---	---	---	---	---	---

 → 

t	t	a	a	b	b	l	l	e	e	a	a	u	u
---	---	---	---	---	---	---	---	---	---	---	---	---	---