

# TP 2 : variables, expressions et instructions

Informatique Fondamentale (IF1)

Semaine du 28 Septembre 2009

Commencez par créer un repertoire nommé `tp2` dans lequel vous travaillerez durant ce TP.

## 1 La classe Deug

Comme vous avez pu le constater, un programme Java commence par des instructions de la forme « `import xxx.yyy.zzz.ttt ;` » qui permettent d'importer des paquetages, c'est-a-dire d'utiliser des ensembles de fonctions predefinies. On utilisera souvent le paquetage `Deug` en mettant au debut des programmes

```
import fr.jussieu.script.Deug;
```

**Exercice 1.** Sans y passer trop de temps, jetez un coup d'œil à la documentation du paquet `Deug`, qui se trouve à l'adresse

<http://www.liafa.jussieu.fr/~yunes/deug/Deug/>

A quoi sert la fonction `Deug.println(int)` ? La fonction `Deug.readInt` ?

## 2 Expressions arithmétiques

**Exercice 2.** Ecrivez un programme `Moyenne` qui lit quatre notes et affiche leur somme et leur moyenne. Modifiez ensuite votre programme pour n'utiliser qu'une seule variable.

**Exercice 3.** Ecrivez un programme `Cercle` qui demande le rayon d'un cercle et affiche son perimetre et sa surface. Vous pourrez vous servir de la constante `Math.PI` dont la valeur est une approximation de  $\pi$ .

**Exercice 4.** Quel est le type et la valeur de chacune des expressions suivantes ?

<code>4 / 3</code>	<code>(4 - 3) * 5</code>	<code>1.3 / 0</code>	<code>(int)2.7</code>
<code>4 / 3.0</code>	<code>117 % 7</code>	<code>(-1.3) / 0</code>	
<code>4 - 3 * 5</code>	<code>0.3 - 0.2</code>	<code>0 / 1.3</code>	
<code>4 - (3 * 5)</code>	<code>0.2 - 0.1</code>	<code>0 / 0</code>	

Ecrivez un programme `ExprArith` qui evalue et affiche la valeur de chacune d'elles pour vérifier vos hypotheses.

**Exercice 5.** Ecrivez un programme `Conversion` qui demande une temperature en degres Celsius et la convertit en degres Fahrenheit. On rappelle la formule

$$f = \frac{9c}{5} + 32$$

ou  $f$  est la temperature en degres Farenheit et  $c$  en degres Celsius. (Pensez au type des variables | ce ne sont pas forcément des entiers.)

**Exercice 6.** Ecrivez un programme `TTC` qui a partir d'un prix hors taxes et d'un taux de TVA calcule le prix toutes taxes comprises correspondant.

### 3 Opérateurs booléens

**Exercice 7.** Quelles sont les valeurs des expressions booléennes suivantes ?

<code>10 &gt; 5</code>	<code>false    (5 != 4)</code>
<code>10 == 5</code>	<code>false &amp;&amp; (5 != 4)</code>
<code>5 == 5</code>	<code>!(30 % 3 == 0)</code>
<code>5 == 11 - 6</code>	<code>0.3 - 0.2 == 0.2 - 0.1</code>

Verifiez vos reponses en ecrivant un programme qui les evalue.

**Exercice 8.** Ecrivez un programme `Bissextile` qui determine si une annee est bissextile. (On rappelle qu'une annee est bissextile si son numero est multiple de 4, sauf si l'annee est aussi multiple de 100 mais pas de 400.)

### 4 Compilation sous XEmacs

Dans le TP precedent nous avons appris a compiler un programme java a l'aide d'une fenetre shell. Cela consistait a ce placer dans le repertoire ou se trouve le fichier a compiler, puis a taper la commande `javac MonProgramme.java`. Il est egalement possible de compiler un programme directement sous XEmacs.

**Exercice 9.** Tapez le programme suivant :

```
import fr.jussieu.script.Deug;

public class ProgrammeBogue{
    public static void main(String[] args){
        int x, x_cube, x_carre;
        Deug.println("Entrez_un_entier");
        x = Deug.readInt();
        x_carre = x * x;
        Deug.println("la_valeur_de_x_carre_est_" + x_carre);
        x_cube = x_carre * x;
        Deug.println("la_valeur_de_x_cube_est_" + x_cube);
    }
}
```

1. Compilez le a l'aide d'XEmacs, pour cela appelez la commande `compile`, en tapant d'abord `M-x` (tenez appuyee la touche *Alt* pour *M*-, appuyez sur la touche *x*, puis relachez la touche *Alt*). Le curseur passe alors dans le *mini buffer*, situe en bas de la fenetre d'XEmacs, tapez alors `compile`.
2. Remplacez `make -k` par la commande de compilation appropriee

```
javac ProgrammeBogue.java
```

puis appuyez sur la touche *entrée*.

3. Pointez sur une erreur, et cliquez sur le bouton du milieu de la souris, que se passe-t-il ?
4. Corrigez les erreurs et recompilez jusqu'à ce que le programme fonctionne.

Dans la suite des TP, c'est à vous de décider si vous préférez compiler directement à partir du shell, ou si vous trouvez que c'est plus pratique à partir d'Emacs.

## 5 Chaînes de caractères

**Exercice 10.** Ecrivez un générateur de lettres de félicitations au charge de TP. Ce programme, appelé *Lettre*, devra demander le prénom d'un charge de TP et votre prénom et devra afficher un message de la forme suivante :

```
Cher chargé_de_TP,  
J'adore vos TPs, ils sont tellement bien !  
Signé étudiant.
```

Modifiez ensuite votre programme pour insister sur la qualité des TP en rajoutant des guillemets autour de « bien », c'est-à-dire en affichant maintenant « ils sont tellement "bien" ! ».

**Exercice 11.** Quel est le type et la valeur des expressions suivantes ?

1. 'a'
2. "pif\npouf"
3. '\\'
4. '\\\"'
5. "aujourd'hui"
6. "dites \"Aaaaah\" !"
7. "un\ttext\tespacié"
8. "deux" + "deux"
9. "deux + deux"
10. 'o' + "k"

Vérifiez vos réponses à l'aide d'un programme.

## 6 S'il vous reste du temps

**Exercice 12.** Calculez le discriminant d'un trinôme  $ax^2 + bx + c$  donné par ses trois coefficients  $a$ ,  $b$  et  $c$ . Affichez aussi s'il admet des racines réelles. En supposant maintenant que tous les trinômes en entrée ont des racines réelles, affichez ces racines.

**Exercice 13.** Ecrivez un programme qui convertit un nombre de secondes en une durée exprimée en heures, minutes et secondes.

**Exercice 14.** Ecrivez un programme qui demande à l'utilisateur trois entiers consécutifs, affiche `true` s'ils sont donnés en ordre croissant, `false` sinon.