

Corrigé des exercices simples sur les boucles

Exercice 3.1.1 déroulement d'une boucle for

Avec un tableau comme celui donné dans l'exemple 4.0, retracez une exécution du programme suivant dans laquelle on entre au clavier la valeur 5.

```
1 class Exo4_1{
2     public static void main(String[] args){
3         int x;
4         Terminal.ecrireString("Un entier svp:");
5         x = Terminal.lireInt();
6         for (int i = 0; i<4; i++){
7             Terminal.ecrireIntln(x+i);
8         }
9         Terminal.ecrireStringln("Fini");
10    }
11 }
```

Voici le tableau qui retrace l'exécution de ce programme :

nb	test	x	i	clavier	écran
3		?	NEP		
4		?	NEP		Un entier svp :
5		5	NEP	5	
6.init		5	0		
6.test	true	5	0		
7		5	0		5
6.modif		5	1		
6.test	true	5	1		
7		5	1		6
6.modif		5	2		
6.test	true	5	2		
7		5	2		7
6.modif		5	3		
6.test	true	5	3		
7		5	3		8
6.modif		5	4		
6.test	false	5	4		
9		5	NEP		Fini

Les affichages seuls de ce programme sont :

```
> java Exo4_1
Un entier svp: 5
5
6
7
8
Fini
```

Exercice 3.1.2 déroulement d'une boucle while

Cet exercice n'est pas corrigé.

Exercice 3.1.3 calculs

1. Écrivez un programme qui affiche la table de multiplication d'un chiffre.
-

Ce chiffre sera entré par l'utilisateur. Par exemple, si le chiffre est 3, le programme affiche:

```
\begin{verbatim}
1_x_3=_3
2_x_3=_6
3_x_3=_9
4_x_3=_12
5_x_3=_15
6_x_3=_18
7_x_3=_21
8_x_3=_24
9_x_3=_27
\end{verbatim}
```

2. Si vous ne l'avez pas déjà fait, modifiez votre programme pour qu'il vérifie que le nombre entré par l'utilisateur est bien un chiffre (c'est à dire un nombre compris entre 1 et 9).
-

```
class Exo4_3_2{
    public static void main(String[] args){
        int chiffre = -1;
        int res;
        while(chiffre<1 || chiffre>9){
            Terminal.ecrireStringln("Entrez le chiffre à multiplier:");
            chiffre = Terminal.lireInt();
            if (chiffre<1 || chiffre>9){
                Terminal.ecrireString("Ce nombre n'est pas compris entre 1");
                Terminal.ecrireStringln("et 9. Recommencez.");
            }
        }
        for (int i=1; i<10; i++){
            res = i*chiffre;
            Terminal.ecrireInt(i);
            Terminal.ecrireString("x" + chiffre + "=");
        }
    }
}
```

```

        Terminal.ecrireIntln(res);
    }
}

```

3. Écrivez un programme qui calcule x^y où x et y sont deux entiers saisis au clavier.

```

class Exo4_3_3{
    public static void main(String[] args){
        int x, y;
        int res=1;
        Terminal.ecrireStringln("Entrez le chiffre a multiplier:");
        x = Terminal.lireInt();
        Terminal.ecrireStringln("Entrez l'exposant:");
        y = Terminal.lireInt();
        for (int i=1; i<=y; i++){
            res = res*x;
        }
        Terminal.ecrireInt(x);
        Terminal.ecrireStringln(" _puissance_ "+y+" _vaut_ "+res);
    }
}

```

4. Écrivez un programme qui affiche la valeur de la fonction x^2 (la fonction qui à un entier associe son carré) pour les dix premiers entiers positifs.

```

class Exo4_3_4{
    public static void main(String[] args){
        int carre;
        for (int i=1; i<=10; i++){
            carre = i*i;
            Terminal.ecrireIntln(carre);
        }
    }
}

```

Exercice 3.1.4 *conversion en dollars*

Le programme suivant est une version du programme Conversion des notes de cours, modifié afin de calculer la conversion en dollars d'une somme en euros saisie au clavier.

```

public class Exo2_1_2 {
    public static void main (String[] args) {
        double euros, dollar, cours;

        Terminal.ecrireStringln("Cours du dollar (valeur de 1 dollar)?");
        cours = Terminal.lireDouble();
        Terminal.ecrireStringln("Somme en euros?");
        euros = Terminal.lireDouble();
        dollar = euros / cours;
        Terminal.ecrireStringln("La somme en francs: "+ dollar);
    }
}

```

```
}
```

Modifiez ce programme pour que pour un cours du dollar donné (et qui ne change pas pendant l'exécution du programme), on puisse convertir plusieurs sommes. Prévoyez un moyen d'arrêter l'exécution du programme.

```
public class Exo4_4 {
    public static void main (String[] args) {
        double euros, dollar, cours;
        char reponse = 'o';

        Terminal.ecrireStringln("Cours_du_dollar_(valeur_de_1_dollar)?_");
        cours = Terminal.lireDouble();
        while (reponse == 'o'){
            Terminal.ecrireStringln("Somme_en_euros?_");
            euros = Terminal.lireDouble();
            dollar = euros / cours;
            Terminal.ecrireStringln("La_somme_en_francs:_"+ dollar);
            Terminal.ecrireStringln("Voulez-vous_faire_une_autre_conversion" +
                                    "_tapez_o_pour_oui_ou_n_pour_non");
            reponse = Terminal.lireChar();
        }
    }
}
```

Exercice 3.1.5 *date correcte*

Écrivez un programme qui saisit une date correcte sous la forme de trois entiers (jour, mois et année). Le programme doit tester si la date est correcte, et si ce n'est pas le cas, doit signaler le type d'erreur rencontrée, puis demander une nouvelle saisie. Le programme finit lorsqu'une date correcte est enfin saisie, avec l'affichage de celle-ci. Dans le cas où le mois de la date est février, votre programme devra calculer si l'année est bissextile. De manière générale, il devra calculer le nombre de jours maximal du le mois de la date saisie, de manière à valider le numéro de jour qui a été saisi.

Réponse : Il n'y a pas de réelle difficulté dans ce programme, si ce n'est de bien organiser les calculs. Par exemple, si l'année et le mois sont correctes, on peut calculer le nombre maximal de jours dans le mois pour cette année, et ensuite utiliser ce nombre pour tester si le numéro de jour donné est correcte. En revanche, cela n'a pas de sens de faire ce calcul si le mois ou l'année sont invalides.

```
class DateCorrecte {
    /* Saisit une date et teste si elle correcte.
       Demande une nouvelle saisie tant que la date
       est incorrecte et signale l'erreur eventuelle
       (1 erreur a la fois).
    */
    public static void main(String [] args) {
        int jour;
        int mois;
        int annee;
        boolean dateOK = false;
```

```

do {
    Terminal.ecrireString("Donnez un jour: ");
    jour = Terminal.lireInt();
    Terminal.ecrireString("Donnez un mois: ");
    mois = Terminal.lireInt();
    Terminal.ecrireString("Donnez une annee: ");
    annee = Terminal.lireInt();
    if (annee <= 0) {
        Terminal.ecrireStringln("L'annee doit etre positive.");
    } else if (mois < 1 || mois > 12) {
        Terminal.ecrireStringln("Le mois doit etre compris entre 1..12");
    } else {
        int nbJours;
        if (mois == 4 || mois == 6 || mois == 9 || mois == 11) {
            nbJours = 30;
        } else if (mois == 2) {
            boolean bissextile =
                (annee % 4 == 0) && !(annee % 100 == 0) || (annee % 400 == 0);
            if (bissextile)
                nbJours = 29;
            else nbJours = 28;
        } else {
            nbJours = 31;
        }
        if (jour < 1 || jour > nbJours) {
            Terminal.ecrireString("Numero de jour incorrect: pour le mois "+mois);
            Terminal.ecrireString(" de l'annee "+annee);
            Terminal.ecrireStringln(", le jour doit etre compris entre 1.."+nbJours);
        } else {
            dateOK = true;
        }
    }
} while (!dateOK);
Terminal.ecrireString("La date saisie est ");
Terminal.ecrireInt(jour);
Terminal.ecrireString("/");
Terminal.ecrireInt(mois);
Terminal.ecrireString("/");
Terminal.ecrireIntln(annee);
}

```
