

1 Recherche de motifs

Exercice 3.

```
import fr.jussieu.script.Deug;
public class Recherche2{
    public static boolean recherche2(int [][] a, int v){
        for (int i=0;i<a.length;i++){
            for (int j=0;j<a[i].length;j++){
                if (a[i][j]==v)
                    return true;
            }
        }
        return false;
    }
}

public static void main(String[] args){
    int [][] t = Prelim.saisie();
    Deug.print("Entrez une valeur : ");
    int v = Deug.readInt();
    if (recherche2(t,v)
        Deug.println("Elle est dans le tableau");
    else
        Deug.println("Elle n'est pas dans le tableau");
    }
}
```

Exercice 4.

Rappelons tout d'abord l'énoncé correct : il s'agit de trouver s'il existe des indices k, l tels que pour tous $i < s.length$ et $j < s[0].length$,

$$a[k+i][l+j] = s[i][j]$$

La solution la plus simple consiste à utiliser une fonction auxiliaire permettant de tester si le tableau s est un sous-tableau de a à une position (k, l) donnée. Attention aux bornes de variation de k et l !

```
import fr.jussieu.script.Deug;
public class Recherche3{
    public static boolean recherche3_aux(int [][] a, int [][] s, int
        for (int i=0;i<s.length;i++){
            for (int j=0;j<s[0].length;j++){
                if (a[k+i][l+j] != s[i][j])
                    return false;
            }
        }
    }
}
```

TP 7 : Tableaux à deux dimensions

Informatique Fondamentale (IF1)

26 décembre 2008

Préliminaires

Exercice 1. Voir l'exercice suivant.

Exercice 2.

```
import fr.jussieu.script.Deug;
public class Prelim{
    public static int [][] saisie(){
        Deug.print("Nombre de lignes : ");
        int k = Deug.readInt();
        Deug.print("Nombre de colonnes : ");
        int l = Deug.readInt();
        int [][] t = new int[k][l];
        for (int i=0;i<k;i++){
            for (int j=0;j<l;j++){
                Deug.print("Entrez le nombre (" + i + ", " + j + ") : ");
                t[i][j] = Deug.readInt();
            }
        }
        return t;
    }
}

public static void affichage(int [][] a){
    for (int i=0;i<a.length;i++){
        for (int j=0;j<a[i].length;j++){
            Deug.print(a[i][j] + " ");
        }
        Deug.println();
    }
}

public static void main(String[] args){
    int [][] t = saisie();
    affichage(t);
}
}
```



```

    return false;
}
public static void main(String[] args){
    Deug.println("Entrez une matrice : ");
    int [][] a = Prelim.saisie();
    if(Carre.carre(a)){
        if(magique(a))
            Deug.println("La matrice est un carre magique!");
        else
            Deug.println("Ce carre n'est pas magique ...");
    }
    else
        Deug.println("Ce n'est pas un carre ...");
}
}

```

Exercice 9. La solution est présentée dans la solution de l'exercice suivant.

Exercice 10.

```

import fr.jussieu.script.Deug;
public class Normal{
    public static int[] histogramme(int [][] a,int n){
        int[] h = new int[n];
        for(int i=0;i<a.length;i++){
            for(int j=0;j<a[i].length;j++){
                if(a[i][j] < n-1){
                    h[a[i][j]] = h[a[i][j]] + 1;
                }
            }
        }
        return h;
    }
}

```

```

public static boolean normal(int [][] a){
    int l = a.length;
    int[] h = histogramme(a,l*l+1);
    for(int k = 1;k<=l*l;k++){
        if(h[k] != 1)
            return false;
    }
    return true;
}

public static void main(String[] args){
}
}

```

```

}
}
Exercice 7.
import fr.jussieu.script.Deug;
public class Diagonale{
    public static int diagonale1(int [][] a){
        int somme = 0;
        for(int j = 0 ; j<a.length;j++){
            somme = somme + a[j][j];
        }
        return somme;
    }
}
public static int diagonale2(int [][] a){
    int somme = 0;
    for(int i = 0 ; i<a.length;i++){
        somme = somme + a[i][a.length-1-i];
    }
    return somme;
}
}
public static void main(String[] args){
    Deug.println("Entrez une matrice : ");
    int [][] a = Prelim.saisie();
    Deug.println("La trace vaut : "+diagonale1(a));
    Deug.println("La somme de l'autre diagonale : "+diagonale2(a));
}
}
}

```

Exercice 8. Dans cette question, et contrairement à ce que l'énoncé précise, il s'agit juste de déterminer si une matrice est un carré magique. Il n'est question de normalité que dans les exercices suivants.

```

import fr.jussieu.script.Deug;
public class Magique{
    public static boolean magique(int [][] a){
        int somme = Diagonale.diagonale1(a);
        for(int k=0;k<a.length;k++){
            if(Somme.ligne(a,k) != somme ||
                Somme.colonne(a,k) != somme)
                return false;
        }
        if(Diagonale.diagonale2(a) == somme)
            return true;
        else
    }
}

```

```

Deug.println("Entrez une matrice : ");
int [][] a = Prelim.saisie();
if(Carre.carre(a)){
    if(Magique.magique(a))
        if(normal(a))
            Deug.println("La matrice est un carre magique normal !");
        else
            Deug.println("La matrice est un carre magique !");
    }
    Deug.println("Ce carre n'est pas magique ...");
}
else
    Deug.println("Ce n'est pas un carre ...");
}
}
...

```