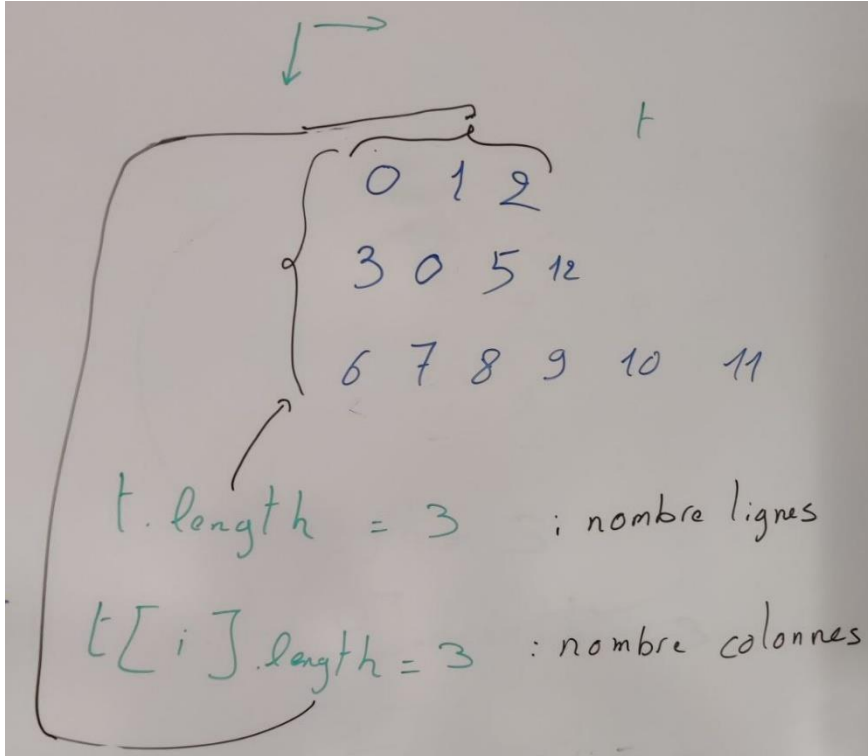


Correction TD8 – Groupe 2

Exercice 1 :

Un exemple de matrice à 2 dimensions :



1-

```
static int max (int[][] tab) {  
    int valMax = tab[0][0];  
    for (int i = 0; i < tab.length; i++) { // lignes  
        for (int j = 0; j < tab[i].length; j++) { // colonnes  
            if (valMax < tab[i][j]) {  
                valMax = tab[i][j];  
            }  
        }  
    }  
    return valMax;  
}
```

2- Avec boucle While :

```
static boolean diagIdentiques (int[][] matrice) {  
    int i = 1;  
    while (i < matrice.length) {  
        if (matrice[0][0] != matrice[i][i]) {  
            return false;  
        }  
        i++;  
    }  
    return true;  
}
```

Avec boucle For :

```
static boolean diagIdentiques (int[][] matrice) {  
    for (int i = 1; i < matrice.length; i++) {  
        if (matrice[0][0] != matrice[i][i]) {  
            return false;  
        }  
    }  
    return true;  
}
```

Exercice 2 :

1-

```
static void affiche (boolean [][] tab){
    for (int i=0; i<tab.length; i++){
        for (int j=0; j<tab[i].length; j++){
            if (tab[i][j]){
                sop("0");
            } else {
                sop(" ");
            }
        }
        System.out.println();
    }
}
```

2-

```
static int nbDoubles (boolean [][] tab){
    int cpt=0;
    for (int i=0; i<tab.length; i++){
        for (int j=0; j<tab[i].length-1; j++){
            if (tab[i][j]==true && tab[i][j+1]==true){
                cpt++;
            }
        }
    }
    return cpt;
}
```

Parcours de la matrice en regardant l'élément suivant

true	false	true
false	true	true

exemple de matrice

```
int cpt=0;
for (int i=0; i<tab.length; i++){
    for (int j=1; j<tab[i].length; j++){
        if (tab[i][j-1]==true && tab[i][j]==true){
            cpt++;
        }
    }
}
```

Parcours de la matrice en regardant l'élément précédent