

Manes Alexandre

15/01/2020

Amplitude des éléments d'un tableau de dimension 2

Sommaire :

- 1).Rappelle de la problématique
- 2).....Algorithme pseudo-code
- 3).....Implémentation en langage Java ou C (code commenté)
- 4).....Jeux de Test
- 5).....Conclusion

Rappelle de la problématique :

Ecrire un algorithme qui remplit un tableau de 5 x 2 entier (tableau à deux dimensions) en liant au clavier chaque élément qui affiche l'amplitude , c'est-à-dire l'écart entre le plus petit et le plus grand élément

ALGORITHME EN PSEUDO-CODE :

Constante : L=5 C=2

Variable : Max , Min , i , j

Initialisation de tab [i][j]

Début :

Pour i allant de 0 a L faire

Pour j allant de 0 a C faire

Ecrire (« Entrer les valeurs du tableau »)

Lire (« tab [i][j] »)

Fin_Pour

Min ← Tab [0][0]

Pour i allant de 0 a L faire

Pour j allant de 0 a C faire

Si Tab[i][j] est inférieur a Min

Alors Min = Tab [i][j]

Fin_Pour

Max ← Tab [0][0]

Pour i allant de 0 a L faire

Pour j allant de 0 a C faire

Si Tab[i][j] est supérieur a Max

Alors Max = Tab [i][j]

Fin_Pour

Amp = Max – Min

Ecrire (« l'émplitude est donc de : » Amp)

FIN

Implémentation en langage Java ou C

```
import java.util.Scanner;
```

```
/**
```

```
*
```

```
* @author formation_gep
```

```
*/
```

```
public class Amlitude_des_elements_dun_tableau_2_dimension {
```

```
/**
```

```
* @param args the command line arguments
```

```
*/
```

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

```
    // déclaration des constante
```

```
    final int L = 5 , C = 2 ;
```

```
    //déclaration et creation de tableau
```

```
    int [][] tab =new int [L] [C];
```

```
    //déclaration des variable
```

```
    int Min , Max , Amp ;
```

```
        Scanner reader = new Scanner(System.in);
```

```
    //Lecture du Tableau , des entrée utilisateur
```

```

for (int i=0 ; i < L; i++){
    for (int j=0 ; j < C ; j++){
        System.out.println("Ecrire un entier:");
        tab[i][j] = reader.nextInt() ;

    }
} //Calcul de Min
Min=tab[0][0] ;
for (int i=0 ; i < L; i++){
    for (int j=0 ; j < C ; j++){
        if (tab[i][j]< Min) {
            Min=tab[i][j];
        }
    }
} //Calcul de MAX
Max=tab[0][0] ;
for (int i=0 ; i < L ; i++){
    for (int j=0 ; j < C ; j++){
        if (tab[i][j]> Max) {
            Max=tab[i][j];
        }
    }
}
}

```

```
//Calcule de Amp et affichage de Amp  
Amp=Max-Min;  
System.out.println("l'amplitude est de "+Amp);  
}  
}
```

Jeux de Test

```
TP_Algo - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)

Output x
TP_Algo - C:\Users\formation_gep\Desktop\TP_Algo\TP_Algo x TP_Algo (run) x
run:
Ecrire un entier:
1
Ecrire un entier:
2
Ecrire un entier:
3
Ecrire un entier:
4
Ecrire un entier:
5
Ecrire un entier:
6
Ecrire un entier:
7
Ecrire un entier:
8
Ecrire un entier:
9
Ecrire un entier:
10
l'amplitude est de 9
BUILD SUCCESSFUL (total time: 16 seconds)

Java Call Hierarchy Finished building TP_Algo (run). 61:2/54:1252 INS
Taper ici pour rechercher
```

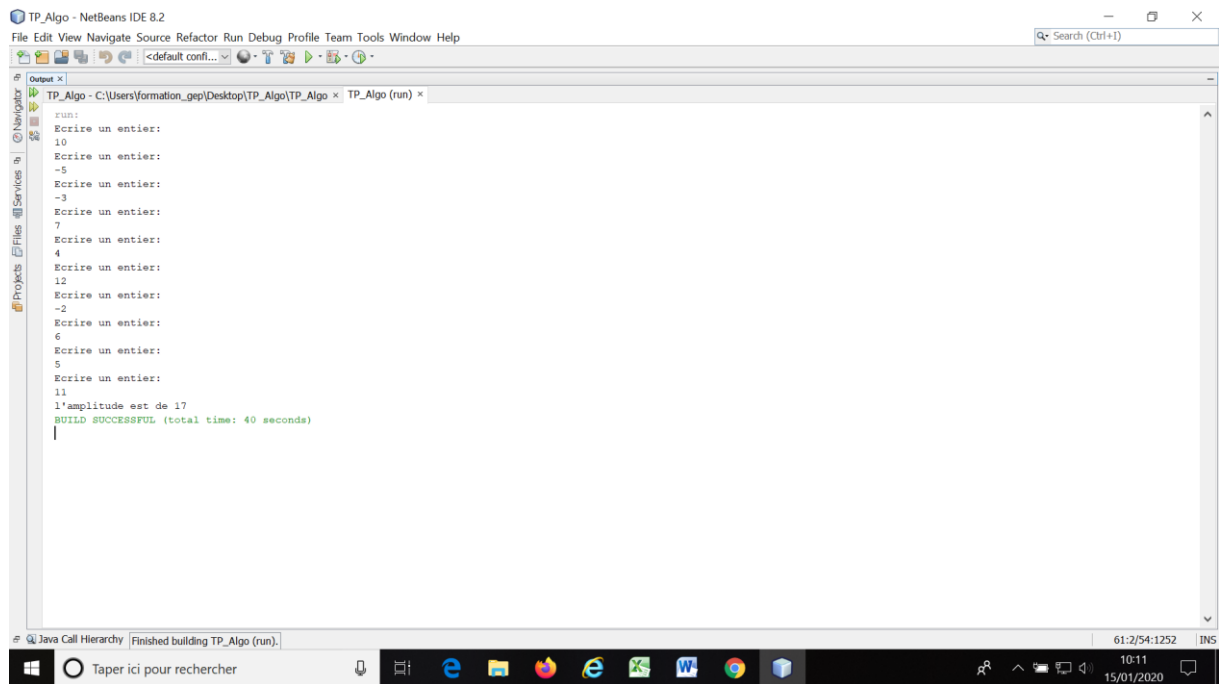
L'emplitude est bien égale a $10-1 = 9$

```
TP_Algo - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)

Output x
TP_Algo - C:\Users\formation_gep\Desktop\TP_Algo\TP_Algo x TP_Algo (run) x
run:
Ecrire un entier:
15
Ecrire un entier:
16
Ecrire un entier:
17
Ecrire un entier:
18
Ecrire un entier:
19
Ecrire un entier:
20
Ecrire un entier:
21
Ecrire un entier:
22
Ecrire un entier:
23
Ecrire un entier:
24
l'amplitude est de 9
BUILD SUCCESSFUL (total time: 23 seconds)

Java Call Hierarchy 61:2/54:1252 INS
Taper ici pour rechercher
```

L'emplitude est bien de $15 - 24 = 9$



L'Amplitude est bien de $12 - (-5) = 17$

Conclusion :

En Conclusion Se Petit programme peut aider à connaître l'écart entre l'entier le plus bas et le plus haut du tableau. Elle peut servir à des comptes pour connaître comme leur croissance sur un nombre de jours, mois, semaine ou année.