

SAE 1.02 - Comparaison Algorithmique

Projet n°1
Rendu n°2

Équipe n°21

Justine BONDU
Jules CHUZEVILLE
Alizéa LEBARON

Exercice 1 :

Tableau résumant nos premiers résultat en millisecondes :

Nom/Type	Byte	Short	Int	Long
Jules	1622	1645	1667	1622
Alizéa	1858	1860	1842	1838
Justine	1858	1791	1798	1775

Comme nous pouvons le remarquer grâce à ce tableau, aucun de nous n'a des résultats similaires avec les mêmes programmes.

Cela peut s'expliquer notamment par les différences de machines. Si l'on veut avoir des résultats plus fiables, il faudrait faire tourner les dits programme au moins un millier de fois.

Tableau résumant la moyenne des résultats en millisecondes :

Nous avons testé les programmes en faisant une moyenne sur 20 tests.

Nom/Type	Byte	Short	Int	Long
Jules	1675	1704	1687	1696
Alizéa	1832	1834	1877	1834
Justine	1924	1767	1765	1794

La logique voudrait que ce soit le byte le plus long à procéder étant donné qu'il soit très souvent retourner à la premières valeurs possible et que cela prend beaucoup de temps.

Et donc que le long soit le plus court (sacré ironie) puisqu'il doit moins souvent retourner à sa première valeur possible.

Cependant nos résultats ne correspondent pas tous à cela, c'est potentiellement du à la différence de nos machines.

Programme du chronomètre seul :

```
/** Chrono
 * date : le 18/11/22
 * @author : Alizée Lebaron
 */

public class Chronometre
{
    public static void main(String[] argv)
    {
        /* Données */
        long    tempsDeb;
        long    tempsFin;
        long    tempsTotSec;

        tempsDeb = System.currentTimeMillis();
        System.out.println(tempsDeb);

        /*Mettre votre itération ici*/

        tempsFin = System.currentTimeMillis();
        System.out.println(tempsFin);

        tempsTotSec = (tempsFin - tempsDeb)/1000;

        System.out.println("Temps en Seconde : " + tempsTotSec);
        System.out.println("Temps en Milliseconde : " + (tempsFin -
                                                                tempsDeb));
    }
}
```

Exécution du programme chronomètre seul (Testé sur la machine d'Alizée) :

```
PS D:\Doc\Université\IUT_Info\sae102> java test
1668778989866 Temps Début
1668778989866 Temps Fin
Temps en Seconde : 0
Temps en Milliseconde : 0
```

Test pour le type "Byte" :

Code:

```
import iut.algo.*;
```

```

/**
 *
 * @author CHUZEVILLE Jules
 */

public class TestByte
{

    public static void main(String[] argv)
    {

        /*-----*/
        /* Données      */
        /*-----*/

        /* Constante      */
        /*-----*/

        final int NB_ITERATION = (int)Math.pow(10,8);

        /* Variables      */
        /*-----*/

        byte    testbyte;
        long    tempsDeb;
        long    tempsFin;
        long    tempsTotSec;

        tempsDeb = System.currentTimeMillis();

        for (int cpt = 0; cpt < NB_ITERATION; cpt++)
        {
            testbyte = (byte) (Math.random() * 50);
        }

        tempsFin = System.currentTimeMillis();

        tempsTotSec = (tempsFin - tempsDeb)/1000;

        System.out.println("Temps en Seconde : " + tempsTotSec);
    }
}

```

```

        System.out.println("Temps en Milliseconde : " + (tempsFin -
tempsDeb));

    }
}

```

Ordinateur de Jules :

```

PS C:\Users\jules> java Int
Temps en Seconde : 1
Temps en Milliseconde : 1622

```

Ordinateur d'Alizéa :

```

PS D:\Doc\Université\IUT_Info\sae102> java TestSec
Temps en Seconde : 1
Temps en Milliseconde : 1858
PS D:\Doc\Université\IUT_Info\sae102> 

```

Ordinateur de Justine:

```

PS C:\Users\justi\Desktop\JUSTINE\Scolaire\BUT_informatique\IUT\TP\s1\s1.02_comparaison_algo> java TestByte.java
Temps en Seconde : 1
Temps en Milliseconde : 1858

```

Test pour le type "short" :

Code:

```

import iut.algo.*;

/**
 *
 * @author CHUZEVILLE Jules
 */

public class TestShort
{

    public static void main(String[] argv)
    {

        /*-----*/
        /* Données      */
        /*-----*/

        /* Constante      */
    }
}

```

```

/*-----*/

final int NB_ITERATION = (int)Math.pow(10,8);

/* Variables      */
/*-----*/

short      testshort;
long       tempsDeb;
long       tempsFin;
long       tempsTotSec;

tempsDeb = System.currentTimeMillis();

for (int cpt = 0; cpt < NB_ITERATION; cpt++)
{
    testshort = (short)(Math.random() * 50);
}

tempsFin = System.currentTimeMillis();

tempsTotSec = (tempsFin - tempsDeb)/1000;

System.out.println("Temps en Seconde : " + tempsTotSec);
System.out.println("Temps en Milliseconde : " + (tempsFin -
tempsDeb));

}
}

```

Testé par l'ordinateur de :

Alizéa :

```

Temps en Seconde : 1
Temps en Milliseconde : 1860

```

Jules :

```
PS C:\Users\jules> java Int
Temps en Seconde : 1
Temps en Milliseconde : 1645
```

Justine:

```
PS C:\Users\justi\Desktop\JUSTINE\Scolaire\BUT_informatique\IUT\TP\s1\s1.02_comparaison_algo> java TestShort.java
Temps en Seconde : 1
Temps en Milliseconde : 1791
```

Test pour le type "int" :

Code:

```
import iut.algo.*;

/**
 *
 * @author CHUZEVILLE Jules
 */

public class TestInt
{

    public static void main(String[] argv)
    {

        /*-----*/
        /* Données */
        /*-----*/

        /* Constante */
        /*-----*/

        final int NB_ITERATION = (int) Math.pow(10,8);

        /* Variables */
        /*-----*/

        byte testbyte;
        long tempsDeb;
        long tempsFin;
        long tempsTotSec;
```

```

        tempsDeb = System.currentTimeMillis();

        for (int cpt = 0; cpt < NB_ITERATION; cpt++)
        {
            testbyte = (int) (Math.random() * 50);
        }

        tempsFin = System.currentTimeMillis();

        tempsTotSec = (tempsFin - tempsDeb)/1000;

        System.out.println("Temps en Seconde : " + tempsTotSec);
        System.out.println("Temps en Milliseconde : " + (tempsFin -
tempsDeb));

    }
}

```

Pour la machine de :

Justine

```

PS C:\Users\justi\Desktop\JUSTINE\Scolaire\BUT_informatique\IUT\TP\s1\s1.02_comparaison_algo> java Int.java
Temps en Seconde : 1
Temps en Milliseconde : 1798

```

Jules

```

PS C:\Users\jules> java Int
Temps en Seconde : 1
Temps en Milliseconde : 1667

```

Alizéa

```

Temps en Seconde : 1
Temps en Milliseconde : 1842

```

Test pour le type “long” :

Code:

```

import iut.algo.*;

/**
 *
 * @author CHUZEVILLE Jules
 */

public class TestLong
{

```

```

public static void main(String[] argv)
{
    /*-----*/
    /* Données      */
    /*-----*/

    /* Constante      */
    /*-----*/

    final int NB_ITERATION = (int) Math.pow(10,8);

    /* Variables      */
    /*-----*/

    byte    testbyte;
    long     tempsDeb;
    long     tempsFin;
    long     tempsTotSec;

    tempsDeb = System.currentTimeMillis();

    for (int cpt = 0; cpt < NB_ITERATION; cpt++)
    {
        testbyte = (long) (Math.random() * 50);
    }

    tempsFin = System.currentTimeMillis();

    tempsTotSec = (tempsFin - tempsDeb)/1000;

    System.out.println("Temps en Seconde : " + tempsTotSec);
    System.out.println("Temps en Milliseconde : " + (tempsFin -
tempsDeb));

}
}

```

Pour la machine de :

Justine

```
PS C:\Users\justi\Desktop\JUSTINE\Scolaire\BUT_informatique\IUT\TP\s1\s1.02_comparaison_algo> java Long.java  
Temps en Seconde : 1  
Temps en Milliseconde : 1775
```

Jules

```
PS C:\Users\jules> java Int  
Temps en Seconde : 1  
Temps en Milliseconde : 1622
```

Alizéa

```
PS D:\Doc\Université\IUT_Info\sae102> java TestSec  
Temps en Seconde : 1  
Temps en Milliseconde : 1838
```