

Exercices comparaisons vitesse IDE

Dans cet exercice, vous allez devoir concevoir un programme permettant de calculer le PGCD de deux nombres de l'afficher et d'afficher également le temps mis dans l'exécution du programme. Utilisez toujours les mêmes nombres pour permettre de comparer de façon égale.

Deux annexes sont fournies expliquants comment utiliser les fonctions permettant de calculer le temps d'exécution.

L'algorithme pour Visual Studio (C#) est le suivant avec n'importe quel nombre entier positif pour X et Y (N'oubliez pas un ReadKey() à la fin du programme pour pouvoir lire la console) :

```
VARIABLE temps <- Stopwatch.DébutNouveau()
```

```
ENTIER a <- X
```

```
ENTIER b <- Y
```

```
SI a est plus grand que 0 ET b plus grand que 0
```

```
    Alors
```

```
        AFFICHER "PGCD(" + a + "," + b + ") = ";
```

```
        TANT QUE a est différent de b
```

```
            Alors
```

```
                SI a est plus petit que b
```

```
                    b <- b - a
```

```
                FIN SI
```

```
            Sinon
```

```
                a <- a - b
```

```
            FIN SINON
```

```
        AFFICHER "PGCD(" + a + "," + b + ") = ";
```

```
    FIN TANT QUE
```

```
    AFFICHER a
```

```
    temps.Arret()
```

```
    VARIABLE tempsMillisecondes <- temps.ElapsedMilliseconds
```

```
    AFFICHER SUR NOUVELLE LIGNE "Temps en millisecondes + tempsMillisecondes "
```

```
FIN SI
```

L'algorithme pour NetBeans et Eclipse (java) est le suivant avec n'importe quel nombre entier positif pour X et Y :

LONG millis_debut <- currentTimeMillis()

ENTIER a <- X

ENTIER b <- Y

Si a est plus grand que 0 ET b plus grand que 0

Alors

AFFICHER "PGCD(" + a + "," + b + ") = ";

TANT QUE a est différent de b

Alors

Si a est plus petit que b

b <- b - a

FIN SI

Sinon

a <- a - b

FIN SINON

AFFICHER "PGCD(" + a + "," + b + ") = ";

FIN TANT QUE

AFFICHER a

VARIABLE millis_fin <- currentTimeMillis()

AFFICHER SUR NOUVELLE LIGNE "Temps en milli secondes + (millis_fin -
millis_debut)"

FIN SI

ANNEXE 1

Les fonctions nous permettant de calculer le temps d'exécution sur en C# sont les suivantes :

System.Diagnostics.Stopwatch.StartNew();

Grâce à cette méthode, un timer va démarrer sur la variable à laquelle elle a été affecté.

var watch = System.Diagnostics.Stopwatch.StartNew();

watch est donc la variable dans laquelle va être stocké le temps.

watch.Stop();

Le timer va être arrêté et **watch** va donc avoir une valeur fixe que l'on pourra afficher de cette façon :

var elapsedMs = watch.ElapsedMilliseconds;

elapsedMs va se voir affecté le temps écoulé en millisecondes et donc le temps d'exécution

ANNEXE 2

Les fonctions nous permettant de calculer le temps d'exécution sur en Java sont les suivantes :

long millis_starttime = System.currentTimeMillis();

millis_starttime va donc prendre comme valeur le temps écoulé au moment où la fonction est appelée, il est donc important de mettre cette ligne au début du programme.

long millis_endtime = System.currentTimeMillis();

millis_endtime va, tout comme **millis_starttime**, prendre la valeur du temps à cet instant, mais cette ligne est placée plus loin dans le programme et donc aura un temps plus avancé.

Pour avoir le temps d'exécution il suffit simplement de soustraire à **millis_endtime** le temps de **millis_starttime** à la fin du programme.