

Nous utilisons la même fonction pour le parallèle et le séquentiel, elle prend un interval en paramètre et fait sont travail.

Le programme mono-para exécute la partie parallèle en premier puis la partie série.

— C++ séquentiel, parallèle

```
./mono-para aabaabb
```

```
aabaabb, 45 4294967268 4294967267 15 19 39 18
```

```
Parallel :
```

```
Time 1.20073 s.
```

```
Trouve aabaabb, 45 4294967268 4294967267 15 19 39 18
```

```
Séquentiel :
```

```
Time 1.11264 s.
```

```
Trouve aabaabb, 45 4294967268 4294967267 15 19 39 18
```

```
Done
```

— OpenCL

```
Entrez la chaine de caractere de longueur 7
```

```
aabaabb
```

```
Wait while processing...
```

```
Time 10.1712 s.
```

```
Le mot de pass trouve est : aabaabb
```

Le programme parallèle est divisé en deux threads, nous avons 4 coeurs sur la machine de test donc le plus proche diviseur de 26 en entier est 2. On obtient donc des résultats similaire en c++ séquentiel et parallèle pour l'intervall suivant : aaaaaa - nnnnnnn

Car, la séparation de l'alphabet se fait comme suit : aaaaaa - nnnnnnn nnnnnnn - zzzzzzz

Donc pour tout ce qui est au dessus de nnnnnnn la partie parallèle est plus rapide d'un facteur d'environ 2 dans notre cas, sinon la partie séquentielle est plus rapide d'un facteur [1.0786, 2.7852] plus le mot de

pas est proche de aaaaaa plus la partie séquentielle à de meilleur résultats.

La partie OpenCL se calcule de la même façon mais divisé en 26 parties.

— C++ séquentiel, parallèle

```
./mono-para aazbbbb
```

```
nnnonnn, 45 4294967268 4294967267 15 19 39 18
```

```
Parallel :
```

```
Time 33.6309 s.
```

```
Trouve nnnnonnn, 45 4294967268 4294967267 15 19 39 18
```

```
Sequentiel :
```

```
Time 32.3821 s.
```

```
Trouve nnnnonnn, 45 4294967268 4294967267 15 19 39 18
```

```
Done
```

— OpenCL

```
Entrez la chaîne de caractère de longueur 7
```

```
aazbbbb
```

```
Wait while processing...
```

```
Time 140.748 s.
```

```
Le mot de pass trouve est : aabaabb
```

— C++ séquentiel, parallèle

```
./mono-para nnnnonnn
```

```
nnnonnn, 45 4294967268 4294967267 15 19 39 18
```

```
Parallel :
```

```
Time s.
```

```
Trouve nnnnonnn, 45 4294967268 4294967267 15 19 39 18
```

Séquentiel :

Time xxx.xxxx s.

%Trouve nnnonnn, 45 4294967268 4294967267 15 19 39 18

%Done

Trop long pas eu le temps de finir

— OpenCL

Entrez la chaîne de caractère de longueur 7

nnnonnn

Wait while processing...

Time 3.77138 s.

Le mot de passe trouve est : nnnonnn

Il y a quelques mots de passe qui ont le même hash ce qui peut causer des problèmes dans la résolution.

— qqrrrt = aaabbbd