Correction du TD N°1

Algorithmes itératifs

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d’opérations » de chaque séquence

Séquence-1 :

```
For i = 1 to N do
        Opération ;
Endfor
```

Nombre d’exécution de l’opération: N fois

La complexité est $O(N)$

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d'opérations » de chaque séquence

Séquence-2 :

```
For i = 1 to N do
    For j = 1 to i do
        Opération ;
    Endfor
Endfor
```

Nombre d'exécution de l'opération: N*N fois

La complexité est $O(N^2)$

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d'opérations » de chaque séquence

Séquence-3 :

```
i=1;
While ( i < N) Do
    i = 2*i;
    Opération ;
Endwhile
```

N=2= 2^1 -> 1 opération

N=4= 2^2 -> 2 opérations

N=8= 2^3 -> 3 opérations

N=16= 2^4 -> 4 opérations

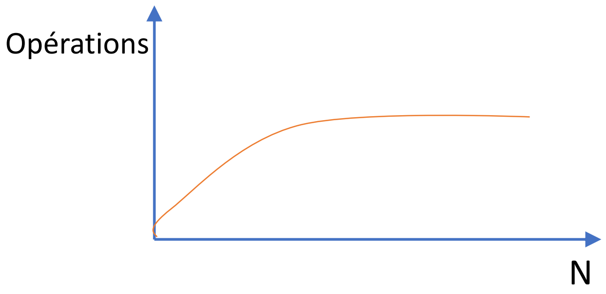
N=32= 2^5 -> 5 opérations

N=64= 2^6 -> 6 opérations

$$N = 2^a$$

a: nombre d'opérations

$$a = log_2(N)$$



La complexité est $O(log_2(N))$

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d'opérations » de chaque séquence

Séquence-4 :

```
For i = 1 To N Do
        J=1;
        While (J < N ) Do
                J = 2 * J;
                Opération;
        Endwhile
Endfor
```

$O(log_2(N))$

$O(nlog_2(N))$

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d'opérations » de chaque séquence

Séquence-5 :

```
i = 1;
While ( i < N ) Do
        i = 2*i;
        For j = 1 to i Do
                Opération;
Endwhile
```

N=2 $=2^1$-> 2 opérations 1

N=3 -> 2+4 opérations 2

N=4 $=2^2$-> 2+4 opérations 2

N=5 ->2+4+8 opérations 3

N=6 ->2+4+8 opérations 3

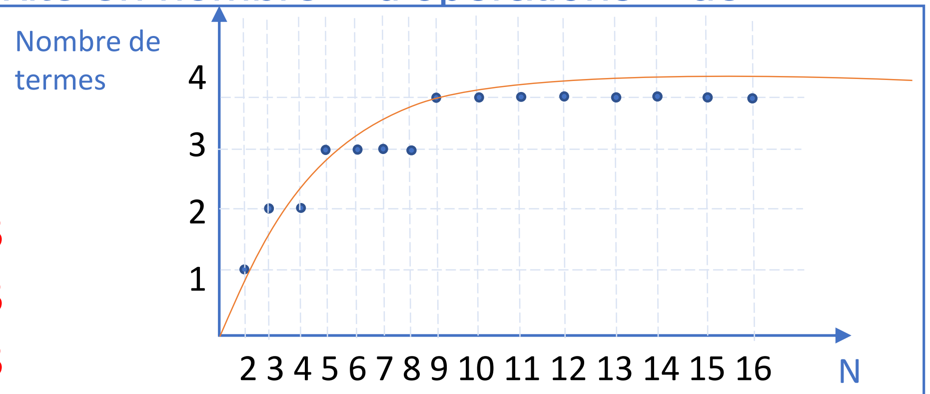
N=7 ->2+4+8 opérations 3

N=8 $=2^3$->2+4+8 opérations 3

N=16 $=2^4$-> 2+4+8+16 opérations 4

N=32 $=2^5$-> 2+4+8+16+32 opérations 5

N $=2^k$-> 2+4+8+16+32+….+N opérations **k:nbr_termes**



suite géométrique: 2+(2*2)+(2*4)+(2*8)+(2*16)+…+(2*N)

- raison q=2
- U_0=2
- nombre de termes=$log_2(n)$

$$U_0 \frac{1-q^{nbr\_termes}}{1-q}$$

$$\Rightarrow 2\frac{1-2^{\log(n)}}{1-2} \quad \cong O(2^{log_2(n)})$$

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d’opérations » de chaque séquence

Séquence-6:

```
i = 1
For j=1 To n do
        i = 2*i
Endfor
For j= 1 to i do
        Opération;
Endfor
```

n=1	->i=2 =>2^1 opérations
n=2	-> i=2*2=> 2^2opérations
n=3	->i=2*2*2=8=> 2^3 opérations
n=4	-> i=2*2*2*2=16 =>2^4 opérations
n=5	-> i=2*2*2*2*2=32 =>2^5 opérations

$O(2^n)$

Complexité de séquences itératives

Exercice 2: Déterminer (en fonction de n à O( ) près) la complexité en nombre « d'opérations » de chaque séquence

Séquence-7:

```
For k = 1 to n do
        i = 1
        For j = 1 to k do
                i = 2*i
        Endfor
        For j = 1 to i do
                Opération
        Endfor
Endfor
```

n=1 ->k=1 => 2 opérations

n=2 -> (k=1,k=2)=> 2+4 opérations

n=3 -> (k=1,k=2,k=3)=> 2+4+8 opérations

n=4 -> (k=1,k=2,k=3,k=4)=> 2+4+8+16 opérations

n=5 -> (k=1,k=2,k=3,k=4,k=5)=> 2+4+8+16+32 opérations

n => $\underbrace{2^1 + 2^2 + 2^3 + 2^4 + \cdots + 2^n}_{\text{suite géométrique de raison 2}}$ opérations

$$U_0 \frac{1 - q^{nbr\_termes}}{1 - q}$$

$$2(2^0 + 2^1 + 2^2 + \cdots + 2^{n-1}) = 2 * 1 * \frac{1-2^n}{1-2} = 2(2^n - 1)$$

$$= 2^{n+1} - 2 \qquad \cong O(2^n)$$