

[ALGO] Complexité des Algorithmes (1)

Par Rhaeven, relecture par Find3r

Problématique

Algorithme : séquence d'instruction qui résout un problème

Problèmes :

- Trouver le minimum d'un tableau de n valeurs
- Trouver un mot dans un dictionnaire de n mots
- Paramètres par une taille

Complexité temporelle d'un algorithme :

$T_{algo}(n)$ = temps pour résoudre un problème de taille n

Faire des choix entre des algos, ou des prédictions sur le comportement des algos avec un n donné.

$$A(n) = \sum_{i=0}^n i = ?$$

On a $A(n) = 1 + 2 + \dots + n - 1 + n$

Donc $2A(n) = (n + 1)n$

Donc $A(n) = \frac{n(n+1)}{2}$

Les sommes

Formule de la somme :

$$\frac{(\text{premier}_{\text{terme}} + \text{dernier}_{\text{terme}}) * \text{nb}_{\text{terme}}}{2}$$

$$\begin{aligned}\sum_{i=0}^n (i+1)^2 &\rightarrow \sum_{i=1}^{n+1} i^2 = \sum_{i=1}^n i^2 + (n+1)^2 \\ &\rightarrow \sum_{i=0}^n (i^2 + 2i + 1) = \sum_{i=0}^n i^2 + \sum_{i=0}^n i + \sum_{i=0}^n 1\end{aligned}$$

$$(n+1)^2 = 2(\sum_{i=0}^n i) + (n+1)$$

$$\frac{(n+1)n}{2} = \sum_{i=0}^n i$$

$$\begin{aligned}\sum_{i=0}^n i^3 &\rightarrow (\sum_{i=0}^n i^3) + (n+1)^3 \\ &\rightarrow \sum_{i=0}^n (i^3 + 3i^2 + 3i + 1) = \sum_{i=0}^n i^3 + 3 \sum_{i=0}^n i^2 + 3 \sum_{i=0}^n i + (n+1)\end{aligned}$$

$$(n+1)^3 = 3 \sum_{i=0}^n i^2 + \frac{3n(n+1)}{2} + \frac{2(n+1)}{2}$$

$$\text{Avec } \frac{3n(n+1)}{2} + \frac{2(n+1)}{2} = \frac{(3n+2)(n+1)}{2}$$

$$\begin{aligned}3 \sum_{i=0}^n i^2 &= (n+1)^3 - \frac{(3n+2)(n+1)}{2} \\ &= \frac{2(n+1)(n+1)^2}{2} - \frac{(n+1)(3n+2)}{2}\end{aligned}$$

$$\begin{aligned}\sum_{i=0}^n i^2 &= \frac{(n+1)}{6}(2(n+1)^2 - (3n+2)) \\ &= \frac{(n+1)}{6}(2n^2 + 4n + 2 - 3n - 2) \\ &= \frac{n(n+1)}{6}(2n+1)\end{aligned}$$

$$\begin{aligned}S(n) &= \sum_{i=0}^n x^i \\ &= \frac{1-x^{n+1}}{1-x} \rightarrow_{x \rightarrow 1} n+1\end{aligned}$$

$$\begin{aligned}S(n) &= x^0 + x^1 + x^2 + \dots + x^n \\ xS(n) &= x^1 + x^2 + \dots + x^n + x^{n+1} \\ (1-x)S(n) &= x^0 - x^{n+1}\end{aligned}$$

Les logarithmes

$$\log_{10}(1234) = a, \text{ bc ou } a = 3 \text{ car } \log_{10}(1000) = \log_{10}(10^3) = 3$$

$$\log_{10}(1234) - 3 = 0, \text{ bc}$$

$$\log_{10}\left(\frac{1234}{1000}\right) = 0, \text{ bc}$$

$$\log_{10}(1,234) = 0, \text{ bc}$$

$$\log_{10}(1,234^{10}) = b, \text{ cou } 1,234^{10} = 8,1875$$

$$\log_{10}(8,1875) = b, c$$

$$\log_{10}(8,1875) = 0, c \text{ car } 8,1875 < 10$$

$$\log_{10}(8,1875^{10}) = c,$$

$$\log_{10}(1353679866,79) = 9, (\text{nb de chiffres devant la virgule} - 1)$$

$$\text{Donc } \log_{10}(1234) = 3,09$$

$$\log_2(1234) = a, \text{ bc}$$

$$\log_2(1234) = 10, \text{ bc car } \log_2(1024) = 10$$

$$\log_2\left(\frac{1234}{1024}\right) = b, c$$

$$\log_2(6,4588) = 2, c \text{ car } \log_2(4) = 2 \text{ et } \log_2(8) = 3$$

$$\log_2\left(\frac{6,4588}{4}\right) = c,$$

$$\log_2(120,48) = 6$$

10 -> décale la virgule de l'autre côté de l'équation

Exemples concrets

```
void fun(int n)
{
    for (int i = 5; i <= n; ++i)
    {
        puts("A");
        for (int j = 10; j < 20; ++j)
            puts("B")
        for (int j = 1; j < i; ++j)
            puts("C");
        for (int j = 10; j < i; ++j)
            puts("D");
    }
}
```

Calculer combien de fois chaque lettre est affichée ?

Avec 10

- $A(10) = 5$
- $B(10) = 50$
- $C(10) = 30$
- $D(10) = 0$

Avec n quelconque

$$A(n) = \sum_{i=5}^{n-1} 1 = \begin{cases} n - 5 & \text{ssi } n > 5 \\ 0 & \text{sinon} \end{cases}$$

$$B(n) = \sum_{i=5}^{n-1} \sum_{j=10}^{19} 1 = \sum_{i=5}^{n-1} 10 = \begin{cases} 10(n - 5) & \text{ssi } n > 5 \\ 0 & \text{sinon} \end{cases}$$

$$C(n) = \sum_{i=5}^{n-1} \sum_{j=i}^{i-1} 1 = \sum_{i=5}^{n-1} (i - 1) \\ = \begin{cases} \frac{(4+n-2)(n-5)}{2} = \frac{(n-5)(n+2)}{2} & \text{ssi } n > 5 \\ 0 & \text{sinon} \end{cases}$$

$$D(n) = \sum_{i=5}^{n-1} \sum_{j=10}^{i-1} 1 = \sum_{i=11}^{n-1} (i - 10) \\ = \begin{cases} \frac{(1+n-11)(n-11)}{2} = \frac{(n-10)(n-11)}{2} & \text{ssi } n \geq 12 \\ 0 & \text{sinon} \end{cases}$$

```
for (int i = 1; i < n; ++i)
    for (int j = 1; j < i; j += 2)
        puts("E");
```

Comment calculer $E(n)$ ici ?

On a $j = 1, 3, 5, 7, 9... < i$

Changement de variable avec :

$$j = 2k + 1 < i \iff 2k < i - 1 \iff k < \frac{i-1}{2} \text{ (partie entiere sup)}$$

$i < x \rightarrow$ On prend la partie entière supérieure de x

$i \leq x \rightarrow$ On prend la partie entière inférieure de x

```
for (int i = 1; i < n; ++i)
    for (int k = 0; k < (i - 1) / 2; ++k)
        puts("E");
```

$$\begin{aligned} E(x) &= \sum_{i=1}^{n-1} \sum_{k=0}^{\lceil \frac{i-1}{2} \rceil - 1} 1 \\ &= \sum_{i=1}^{n-1} \lceil \frac{i-1}{2} \rceil \\ &= \sum_{i=0}^{n-2} \lceil \frac{i}{2} \rceil = \sum_{i=1}^{n-2} \lceil \frac{i}{2} \rceil \\ &= 1 + 1 + 2 + 2 + 3 + 3 + \dots + \lceil \frac{n-2}{2} \rceil \end{aligned}$$

Si n est pair, tous les termes sont en double

$$E(n) = 2 \sum_{i=1}^{\frac{n-2}{2}} i = (1 + \frac{n}{2} - 1)(\frac{n}{2} - 1) = \frac{n(n-2)}{4}$$

Si n est impair

$$E(n) = E(n-1) + \lceil \frac{n-2}{2} \rceil = \frac{(n-1)(n-3)}{4} + \frac{n-1}{2} = \frac{(n-1)^2}{4}$$