

DIVISER POUR RÉGNER : SYNTHÈSE DE COURS

1) Principe :

« Diviser pour régner » est une technique de programmation qui consiste à diviser un problème complexe en problèmes plus petits qui sont en conséquence plus simples à résoudre.

On a l'habitude de distinguer 3 étapes :

- 1) **Diviser** : Le problème initial est divisé (souvent récursivement) en un certain nombre de sous-problèmes indépendants, plus petits et plus faciles à résoudre.
- 2) **Régner** : On résout ces sous-problèmes.
- 3) **Combiner** : On détermine la solution au problème initial à partir des solutions des sous-problèmes.

2) Exemple classiques :

a) Recherche par dichotomie

Cet algorithme rencontré en première permet de chercher très efficacement si une valeur est dans une liste triée.

Principe : A chaque étape, on divise par deux la taille de la plage de recherche.

On compare la valeur cherchée avec le nombre qui est au milieu de la plage de recherche. Si elle est plus grande, on ne garde que la moitié droite de la plage de recherche. Sinon, on ne garde que la moitié gauche.

Ex : Cherchons si 11 appartient à [1, 7, 8, 10, 11, 15].

À chaque étape, on cherche 11 dans la liste :

Étape						
1	1	7	8	10	11	15
2				10	11	15
3				10	11	
4					11	

Complexité : $\log(n)$.

b) Exponentiation rapide

Principe : On veut calculer x^n .

- Si n est pair, plutôt que de faire de nombreuses multiplications ($x^n = x \times x \times x \times \dots \times x$), on se ramène en une seule multiplication à un problème deux fois plus petit : $x^n = x^{\frac{n}{2}} \times x^{\frac{n}{2}}$.
- si n est impair, on se ramène en une multiplication au cas où n est pair : $x^n = x \times (x^{n-1})$.

Ex : On cherche à calculer x^9 .

Étape	Calcul	Nombre de multiplications
1	$x^9 = x \times x^8$	1
2	$x^8 = x^4 \times x^4$	1
3	$x^4 = x^2 \times x^2$	1
4	$x^2 = x \times x$	1

Complexité : $\log(n)$.

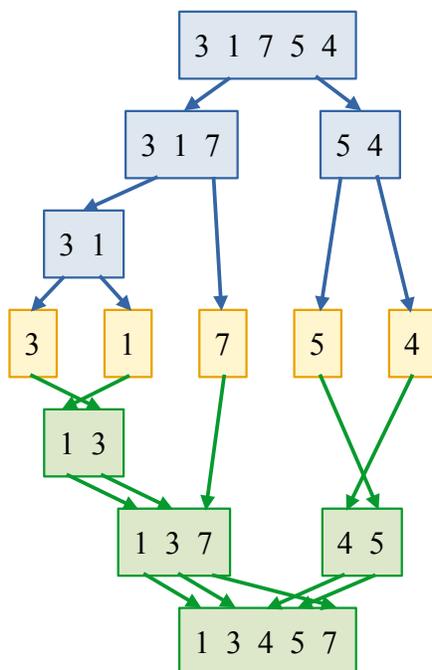
c) Tri fusion :

Principe : On veut trier une liste et on s'appuie sur le fait qu'il est rapide de concaténer deux listes triées en une seule liste elle-même triée.

On commence donc par couper la liste de départ en deux et on continue récursivement jusqu'à obtenir des listes contenant un seul élément. Or une liste contenant un seul élément est triée ! Il ne reste plus qu'à concaténer récursivement les sous-listes triées jusqu'à la liste triée finale.

Pour mieux comprendre : [Explications en vidéo](#) et [animation](#) par Frédéric Boissac.

Ex : On veut trier la liste : [5, 1, 8, 3, 2]



Complexité : $n \times \log(n)$.

d) Quart de tour d'une image :

Principe : On choisit une image carré dont le nombre de pixels de côté est une puissance de 2. Pour effectuer une rotation d'un quart de tour de cette image, on peut diviser l'image en 4, effectuer une permutation circulaire des quatre quadrants, puis appliquer récursivement le même processus à chaque quadrant, jusqu'à obtenir des quadrants contenant un seul pixel.

Une petite vidéo vaut mieux qu'un long discours : <https://www.youtube.com/watch?v=OXo-uzzD4Js>

Ex : Une image étant un tableau de pixels, cela donne sur une image de 8×8 pixels :

Départ (tableau 8 x 8)								1ère étape								2ème étape								Fin							
1	2	3	4	5	6	7	8	33	34	35	36	1	2	3	4	49	50	33	34	17	18	1	2	57	49	41	33	25	17	9	1
9	10	11	12	13	14	15	16	41	42	43	44	9	10	11	12	57	58	41	42	25	26	9	10	58	50	42	34	26	18	10	2
17	18	19	20	21	22	23	24	49	50	51	52	17	18	19	20	51	52	35	36	19	20	3	4	59	51	43	35	27	19	11	3
25	26	27	28	29	30	31	32	57	58	59	60	25	26	27	28	59	60	43	44	27	28	11	12	60	52	44	36	28	20	12	4
33	34	35	36	37	38	39	40	37	38	39	40	5	6	7	8	53	54	37	38	21	22	5	6	61	53	45	37	29	21	13	5
41	42	43	44	45	46	47	48	45	46	47	48	13	14	15	16	61	62	45	46	29	30	13	14	62	54	46	38	30	22	14	6
49	50	51	52	53	54	55	56	53	54	55	56	21	22	23	24	55	56	39	40	23	24	7	8	63	55	47	39	31	23	15	7
57	58	59	60	61	62	63	64	61	62	63	64	29	30	31	32	63	64	47	48	31	32	15	16	64	56	48	40	32	24	16	8

Complexité : Algorithme inefficace mais tellement spectaculaire !