ALGO - Janvier 2019 - PROMO 2021

1. Complexité

* pire
* meillleure
* moyenne
* **amortie**: dépend des cas moyen (exemple: realloc pour le insert tout les n \* 2)

2. Algorithme de tris

a. Tris comparatif Pire cas Meilleur cas

* à bulle Θ(n2) Θ(n2)
* sélection Θ(n2) Θ(n2)
* insertion Θ(n2) Θ(n)
* fusion (ou merge sort) Θ(nlog(n)) Θ(nlog(n))
* par tas (ou heap sort) Θ(nlog(n)) Θ(nlog(n))
* quicksort Θ(nlog(n)) Θ(n2)
* introspectif Θ(nlog(n)) Θ(nlog(n))

(= quick sort, si nb recursion > Klog(n) on tri

le tableau restant avec tri par tas)

b. Tris non comparatifs

* counting sort Θ(n) Θ(n)
* radix sort Θ(n) Θ(n)

c. Propriété

* **Stable** préserve l’ordre des éléments égaux
  + insert sort
  + merge sort
  + counting sort
* **En** **place** consomation mémoire < Θ(nlog(n))
  + tous sauf merge sort

3. CG : quelques algo vu en cours

a.

* **Karatsuba**: multiplication de polynôme
* **Strassen**: multiplication matricielle
* **Diviser pour régner** : diviser un problème en sous problème que l’on sait résoudre
  + merge sort
  + quick sort
  + recherche dichotomique
  + Karatsuba

Leur complexité se calcule avec **le théorème général**.

b. Programmation dynamique

Def: méthode algorithme pour résoudre des problèmes d’optimisations - même mécanisme que

diviser pour régner. Ex: fibonacci : pour calculer n, on calcule n - 1, jusqu’à 1, qu’on connaît.

* Distance de Levenshtein
* Chaîne de multiplication matricielle
* Problème sac à dos (problème d'optimisation combinatoire)

Source:

Wiki de chacun des algo

<http://www.dil.univ-mrs.fr/~gcolas/algo-licence/slides/tris.pdf>

<https://fr.wikipedia.org/wiki/Programmation_dynamique>

<https://fr.wikipedia.org/wiki/Probl%C3%A8me_du_sac_%C3%A0_dos>

<https://epiportal.com/ING/Module%20IF%20-%20Sciences%20de%20base%2C%20Informatique%20Fondamentale/ALGO%20-%20Algorithmie/fiches/refcard.pdf>

(arbre rouge noire en moins, pas au programme)