

# Diviser pour régner et programmation dynamique

Programme de colle

Semaine 24 (24 mars 2025 – 29 mars 2025)

## Programme

### Diviser pour régner

- Théorème borne inférieure des tris par comparaison.
- Exemple : tri fusion. Implémentation en C et OCaml. Analyse de complexité.
- Résolution d'équations de complexité de la forme  $T(n) = rT(n/c) + f(n)$  (avec  $r$  et  $n$  entiers tels que  $r > 0$  et  $c > 1$ ).
- Principe général de la méthode diviser pour régner.
- Exemple : algorithme de Karatsuba. *L'idée de découper un entier en partie haute et partie basse est exigible. Toutefois, l'astuce de calcul qui permet de se réduire à 3 multiplications au lieu de 4 n'est pas exigible.* Analyse de complexité.
- Exemple : tri rapide. Partitionnement (drapeau hollandais). Implémentation en C. Analyse de la complexité dans le pire des cas. Astuce de choix du pivot pour (probablement) éviter le pire des cas.

### Rencontre au milieu

- Exemple pour SubsetSum.

### Programmation dynamique

- Récursivité et redondance des appels en cas de chevauchement de sous-problèmes. Effet sur la complexité temporelle.
- Exemple sur les pyramides.
- Méthode de haut en bas / mémoïsation.

- Exemple sur les pyramides. Analyse de complexité temporelle *et* spatiale.
- Méthode de bas en haut.
- Exemple sur les pyramides.

*L'optimisation de la mémoire parfois permise par la méthode de bas en haut n'a pas encore été vue. La reconstruction d'une solution optimale non plus.*

### Impératif en OCaml

- Les tableaux OCaml ainsi que les boucles ont été introduites. Les références et les enregistrements mutables n'ont pas encore été vus.

### Et plus

Tout ce qui a été fait depuis le début de l'année est au programme. De plus, certains exercices peuvent tout à fait s'écarter du programme pour tester votre réactivité face à l'inconnu.

## Questions de cours

Toute colle commencera par une de ces questions de cours, notée sur 10/20.

Une tolérance vis-à-vis de la syntaxe pourra être appliquée sur les questions d'implémentations si l'élève n'est pas sur machine.

- Implémentation du tri fusion : toute partie du code (division/fusion/tri) choisie par le colleur, en C ou en OCaml.
- Résolution du problème du drapeau hollandais en pseudo-code ou en C.
- Pseudo-code d'une solution mémoisée en programmation dynamique (de haut en bas). Explication de pourquoi (et quand) c'est plus efficace qu'une solution récursive naïve.