

## Programme de colle 1 : du 31/01/2022 au 11/03/2022

Langage : OCaml

### Notions de cours :

- Programmation impérative.
- Analyse de programmes impératifs (terminaison, correction, complexité).
- Programmation récursive.
- Analyse de programmes récursifs (terminaison, correction, complexité).
- Exemples d'algorithmes classiques : exponentiation rapide, recherche dichotomique, tri par insertion, tri par sélection, tri à bulles.
- OCaml : type 'a ref, type 'a array, type 'a list, types enregistrement / énumérés / inductifs.
- Structures linéaires usuelles : pile, file, tableau redimensionnable.
- Diviser pour régner : méthode algorithmique, analyse de la complexité.
- Exemples classiques : tri fusion, tri rapide.
- Structure d'arbre.
- Arbres en OCaml (arbres binaires, arbres binaires entiers, arbres d'arité quelconque).
- Parcours d'arbres en OCaml : parcours en profondeur (préfixe, infixé, postfixé), parcours en largeur.

### Notions avancées (vues en cours mais pas encore en TP) :

- Structure de tas, implémentation d'une file de priorité via un tas, tri par tas.
- Arbres binaires de recherche, implémentation d'un dictionnaire via un ABR.
- Rotations dans un ABR, arbres AVL, arbres bicolores (rouge-noir).