

1 Thématiques informatiques pouvant être abordées

- Cycles dans les graphes
- Boucles dans un programme :
 - * terminaison
 - * logique de Hoare
 - * décidabilité (problème de l'arrêt)
- Structures de données cycliques et applications :
 - * listes chaînées circulaires (tas de fibonacci)
 - * files
- Algorithme de Floyd, détection de cycle
- Comportements cycliques dans les langages formels, automates finis
- Comportements cycliques dans un automate cellulaire, jeu de la vie
- Étude de jeu à deux joueurs infini

2 Références pouvant être utilisées

2.1 Algorithmique générale

- Algorithms (J. Erickson)
- Algorithms (Dasgupta, Papadimitriou, Vazirani)
- Éléments d'algorithmique (Beauquier, Berstel, Chrétienne)
- Informatique MP2I/MPI (Balabonski et al.), disponible au CDI
- Algorithmique (Cormen et al.)

2.2 Algorithmique avancée

- Approximation algorithms (Vazirani)
- Randomized algorithms (Motwani, Raghavan)
- Probability and computing (Mitzenmacher, Upfal)

2.3 Graphes

- Graph Theory (Bondy, Murty)

2.4 Logique

- Introduction à la logique (David, Nour, Raffali)

2.5 Théorie de la complexité

- Complexité algorithmique (Périfel)
- Computational complexity : a modern approach (Arora, Barak)

2.6 Théorie de la calculabilité (dont automates)

- Automata theory, languages, computation (Hopcroft, Motwani, Ullman)
- Introduction to the theory of computation (Sipser)
- Langages formels, calculabilité et complexité (Carton)

2.7 Compilation

- Compilers : principles, techniques and tools (appelé aussi *Dragon book*) (Aho, Sethi, Ullman)

2.8 Concurrency, systèmes d'exploitation

- The little book of semaphores (Downey)
- The art of multiprocessor programming (Herlihy, Shavit)
- Operating systems : three easy pieces (Arpaci, Dusseau)

2.9 Apprentissage automatique

- Artificial intelligence : a modern approach (Russel, Norvig)

3 MCOT

Document à téléverser au plus tard fin janvier. À faire le plus tôt possible pour se faciliter la suite du travail.