

## Distanciel N°3

### Consignes

Le travail en distanciel de cette année consiste à programmer une fonction de manipulation d'un automate. A partir d'une bibliothèque d'outils de manipulation d'automates, comportant un grand nombre de fonctions qui ne sont pas encore implémentées, vous devrez développer en projet de TP les fonctions permettant de standardiser et normaliser un automate. Puis dans le cadre du distanciel vous aurez à vous occuper des fonctions permettant de créer l'automate inverse (transposer) et l'automate mis à l'étoile d'un automate donné. En plus de ce développement, vous devrez proposer un jeu d'essai permettant de tester tous les cas possibles.

### Objectifs du travail :

- comprendre la manipulation des automates d'état fini : en particulier notion d'inverse, de mise à l'étoile ;
- comprendre et reprendre une implémentation en Java ;
- s'inspirer de cette implémentation pour étendre le framework.

### Compétences développées :

- être capable d'aborder une partie de cours en autonomie ;
- être capable de s'exercer et se corriger en autonomie ;
- être capable de comprendre un framework simple en Java ;
- être capable de s'inspirer du framework pour l'étendre.

### Tâches à effectuer :

- ~~1. (Individuel) Relire et comprendre le cours sur la manipulation des automates finis disponible sur e-miage.~~
- ~~2. (Individuel) S'exercer et se corriger avec les exercices proposés par e-miage.~~
- ~~3. (Individuel) Rédiger l'exercice 7 du TD et le rendre via le dépôt madoe avant le 24 novembre;~~
- ~~4. (Individuel) exercice "Jeux d'essai" : proposer plusieurs automates permettant de tester la suite du travail. Chaque automate de ce jeux d'essai doit couvrir un cas possible et doit être différent de l'exemple du cours. L'ensemble des automates doit couvrir tous les cas possibles. Chaque automate doit être justifié (cas ou difficulté mis en valeur).~~
5. (Individuel) récupérer le framework Java JFSM (<https://github.com/edesmontils/JFSM>)
6. (Individuel) comprendre le framework et l'utiliser sur quelques exemples d'automates (par exemple pour tester les propositions de la tâche 4). Tester les différentes classes et méthodes.
7. (Binôme) Implémenter les méthodes "transpose()" et "etoile()" dans la classe "JFSM/Automate.java". Il est possible d'ajouter des méthodes complémentaires qui pourraient être utiles pour l'efficacité et/ou la lisibilité du code.

### Travail à rendre :

- Exercice, images d'automates dans un document PDF, vous pouvez utiliser JFLAP ou faire les schémas à la main puis les scanner.
- un document PDF de 5 pages max. décrivant le développement (architecture, approche, difficultés, solutions trouvées...)
- dans une archive ZIP : la classe Automates avec les méthodes que vous avez développées et au moins deux programmes utilisant la classe Automate utilisant votre développement. Mettre vos noms en commentaire au début des fichiers.

Nous sommes bien entendu à votre disposition si vous avez des questions sur le distanciel. Vous pouvez nous contacter par mail, en parler en fin de séance de TD, TP ou utiliser le forum sur madoc.

### Livrables & Notation :

- ~~Tâche 1, 2 et 3 : exercice 7 (3 pts ; individuel) (à déposer sur une zone de dépôt madoc pour le 24 novembre~~
- ~~Tâche 4 : exercice "Jeux d'essai" (3 pts ; Individuel) (à déposer sur une zone de dépôt Madoc) pour le 8 décembre~~
- Tâche 5,6,7 : Doc PDF (7 pts ; Binôme) pour le 22 décembre sur une zone de dépôt Madoc
- Tâche 5,6,7 : Code (7 pts ; Binôme) pour le 22 décembre sur une zone de dépôt Madoc

### Remarques :

- le non respect des consignes entrainera un 0 pour la tâche concernée ;
- un binôme est composé de deux personnes, les autres configurations ne sont pas autorisées, sauf cas exceptionnel validé par l'enseignant.