

Sujets Projet Note B (30%)  
Algorithmique Avancée  
2021-2022  
Nicolas Loménie (élaboré avec Sylvain Lobry)

La note A (20%) correspond à votre compréhension théorique des algorithmes étudiés et en particulier applicables à des structures de graphes de données. C'est une préparation exemplaire à l'examen final (50%) (Cette note correspond au Rendu Toussaint Section A des TPs/TDs + Rendu Noël Section A des TPs/TDs).

La note B correspond à votre capacité d'implémentation et de résolution de problèmes. Elle est pédagogiquement complémentaire de la compréhension théorique des algorithmes vus en cours. Elle est constituée de deux parties :

- Les rendus de la Toussaint et de Noël correspondant aux sections B des TPs/TDs (sur 10)
- Une deuxième note sur 10 expliquée ci-dessous. Vous aurez le choix entre 2 scénarii en fonction de vos projets d'étudiants ou professionnels (en surlignage).

Les documents nécessaires se trouveront ici <http://helios.mi.parisdescartes.fr/~lomn/Cours/AV/Projet/> .  
Même opératoire que pour la première session. La semaine 8 commence le 8 novembre 2021.

*Rendu 1 : Premier jet : fin de la Semaine 11, Dimanche 5 décembre 2021 23:59*

*Rendu 2 : Rendu final avec projet : fin de la Semaine 13 Dimanche 19 Décembre 2021 23:59*

*Rendu 3 (optionnel) : Amélioration possible : Lundi 3 Janvier 2022 23:59*



## **- Partie Commune**

Vous rendez dans un format à part les exercices d'implémentations qui correspondent aux parties B des TD d'Algorithmique Avancée du semestre. Vous ajoutez une analyse de la complexité de vos algorithmes éventuellement améliorés par vos choix d'implémentation ou de structures intermédiaires comme les tas etc. Vous obtiendrez une note sur 10 correspondant à ce rendu.

Tous les codes java doivent être exécutables sur une machine de l'UFR (Linux avec la commande de compilation détaillées et un README.txt qui explique le fonctionnement du code pour l'exécuter). La clarté d'organisation de vos fichiers dans une arborescence facile à naviguer sera prise en compte. Egalement tout document qui explique l'usage. Enfin les commentaires devront être en Français et personnalisés.

De même pour la suite du mini-projet (Scenario 1 et 2), les codes Python et Java doivent tourner sur les machines de l'UFR (Linux et commandes associées pour l'usage). Une attention pour la notation sera portée sur l'interfaçage graphique et les considérations d'optimisation éventuelles.

## **- Scénario 1 :**

Vous faites évoluer le code fourni (choix du langage Python ou Java indifférent) qui résout le problème du labyrinthe. Vous modéliserez le problème comme la recherche d'un chemin de résolution dans un graphe d'états par une stratégie A\*. Une première heuristique que vous verrez en TP (semaine 7 et 8) sera la distance de la cellule courante à la sortie. Il faudra définir une meilleure heuristique et comparer le résultat. Dans un premier temps le Feu sera assimilé à un mur et donc statique (cas du TP). Puis pour les plus aventureux, vous pourrez réfléchir à une solution pour le cas où le feu avance comme dans l'énoncé. Vous obtiendrez une note sur 10 correspondant à ce rendu.

Fichiers fournis : ***sujetLabyrinthe.pdf*** avec une résolution naïve en C ***labyrinthe.c***.

*Ressources : concernant A\*, vous aurez un TP sur ce sujet en Java s'étendant sur deux semaines. Vous pouvez donc avancer sur l'infrastructure de votre programme et commencer à réfléchir d'ici là si vous le souhaitez à la problématique A\* ([https://en.wikipedia.org/wiki/A\\*\\_search\\_algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm) ou <http://gdac.uqam.ca/inf4230/diapos/04-recherche-heuristique.pdf>) mais en sortie de ces deux TP dédiés à cet algorithme vous aurez une bonne base de travail pour votre rendu final.*

## **- Scénario 2 :**

Vous traitez l'épreuve d'informatique d'entrée aux ENS/Polytechnique. Vous obtenez une note sur 10 correspondant à ce rendu. Vous avez le choix d'implémentation en Python ou Java (originellement en Caml dans le sujet). L'épreuve est censée durer 4h pour évaluer le travail en mode maîtrise du sujet. Il sera pris en compte de la difficulté éventuelle de certains exercices si vous justifiez pour un nombre limité d'entre eux leur non-résolution. En conséquence, il n'est pas nécessaire de traiter tous les points pour avoir la note maximale. Voir le fichier ***2018\_mp\_sujet\_infoENSPolytechnique.pdf***

*Ce scénario est en particulier intéressant pour ceux qui se destinent à tenter des concours d'entrée en Ecole d'ingénieurs. Ou simplement pour se comparer à des étudiants qui suivent des classes préparatoires aux Grandes Ecoles.*

Tout plagiat sera sanctionné. Vous devez dans tous les cas utiliser vos propres formulations, reformulations et codes commentés en citant vos sources.