**Rapport du projet d’Info 4A**

# Explication sur ma génération aléatoire de labyrinthe « intéressant »

Ce jour l\_a vous remettrez un rapport papier. Pour ce document, les consignes strictes de

pr\_esentation sont : au moins 15 pages, police 12pts maximum, marges 2cm maximum. Le rapport doit

comporter au moins les \_el\_ements suivants :

| une analyse fonctionnelle du sujet, pr\_ecisant, entre autre, toutes les r\_egles de fonctionnement

de votre application et un decoupage du probl\_eme en sous-probl\_emes (pour aboutir aux grandes

lignes de l'architecture logicielle), c'est-\_a-dire les classes les plus importantes et leurs liens ;

| une description des structures de donn\_ees envisag\_ees et retenues ;

| la sp\_eci\_cation des classes principales et des m\_ethodes essentielles ;

| l'architecture logicielle d\_etaill\_ee de votre application, c'est-\_a-dire, dans le cadre du mo-

dule Info4B, une conception en couches fonctionnelles \_a l'image de ce qui est r\_ealis\_e dans

l'architecture d'un syst\_eme d'exploitation. Pour chaque couche vous sp\_eci\_erez les classes qui

impl\_ementent les services de la couche fonctionnelle ;

| une description des algorithmes principaux ;

| un jeu de tests montrant que votre programme fonctionne.

Il est conseill\_e de r\_ediger le rapport en utilisant LATEX car il permet d'inclure facilement des

extraits de code source avec une mise en \_evidence des \_el\_ements syntaxiques. On trouve des applica-

tions permettant de r\_ediger collaborativement des documents LATEX (https://fr.sharelatex.com/,

https://fr.overleaf.com/ ou sur le serveur iemgit accessible par VPN en demandant au pr\_ealable

l'activation d'un compte)

Compléments concernant le rapport de projet. En plus de ce qui est spécifié dans le document décrivant les sujets, le rapport de projet de Info42 peut être organisé comme suite :

1. **Rappel du sujet** : expliquez ce que vous en avez retenu, vos adaptations, vos hypothèses, dégager les fonctionnalités principales

2. **Analyse** :

2.1 **Donner l'architecture générale de votre application** : organisation des threads, des communications, faites des schémas

2.2 Pour chaque fonctionnalité, expliquer comment vous la réalisez, préciser les algorithmes que vous allez utiliser, mettez en évidence comment vous résolvez les problèmes de concurrence, de communication (protocole), les autres concepts des systèmes d'exploitation que vous utilisez.

3. **Implémentation** : Décrire les packages, les classes et leur organisation Ne pas mettre tout le code mais des extraits commentés.

4. **Tests et utilisation** Expliquer comment fonctionne votre programme au moyen d'exemples, de copies d'écran etc.

5. Conclusion, évolutions, améliorations

6. Bibliographie (livres ou sites web que vous avez consulté, utilisé ou code que vous avez repris)

7. Annexes : par exemple le code complet du projet.