

UE Projet Python pour la DLMI 2020-2021, session 2

L'UE vise à :

- s'initier à la programmation événementielle
- réaliser des algorithmes
- réaliser une application graphique
- améliorer sa connaissances du langage Python et sa compétence en programmation
- utiliser la programmation objet

Plus précisément, il est attendu de coder une application graphique en Tkinter (aucun autre toolkit n'est envisagé). Selon le sujet, en particulier pour un jeu de type puzzle, l'application graphique pourra nécessiter le codage d'un solveur. La pertinence et la variété des algorithmes mis en oeuvre font alors partie intégrante du sujet.

Il est attendu que toutes les possibilités qu'offre le langage Python soient utilisées, dans la mesure où elles sont adaptées à la résolution du problème. En particulier, la programmation objet pourra être pleinement utilisée, en particulier si elle permet une meilleure lisibilité ou une meilleure extensibilité du programme. La partie graphique en Tkinter sera obligatoirement écrite en utilisant des classes (mais pas forcément dérivant de widgets du toolkit).

De même, il faudra, si c'est pertinent, utiliser les nombreuses structures de données que Python met à disposition (listes bien sûr mais aussi ensembles, files de priorités, dictionnaires, etc).

Il est absolument indispensable qu'il y ait découplage entre la **partie vue** et la **partie métier** de l'application (comme pour un patron MVC mais sans pour autant implémenter un MVC).

L'application sera probablement découpée en plusieurs modules Python.

Soutenance et évaluation

L'équipe dépose son travail sur Moodle sous forme d'un unique fichier zip ou alors envoie par mail un lien vers le fichier zip. Pour l'instant, la date limite de rendu est fixée au dimanche 20 juin 2021. L'évaluation aura lieu lors d'un examen de TP de 3 heures. Vous serez amené à répondre à des questions

- sur la conception de votre projet
- sur les algorithmes mis-en oeuvre
- de maîtrise du langage Python
- de maîtrise des techniques de base de la programmation graphique (événements, widget, callbacks)

En cas de doute sur la maîtrise des techniques, il pourra être demandé de coder des questions simples de programmation graphique (petite animation, utilisation de widget).

Les membres d'une même équipe pourront être interrogés séparément et il pourra être tenu compte dans la note d'un déséquilibre de participation ou de maîtrise.

L'évaluation prendra en compte

- la qualité de la production graphique réalisée : adéquation au projet, absence de bugs, fonctionnalités, usabilité, esthétique, originalité ;
- la qualité du codage Python et des algorithmes écrits ;
- le suivi du projet : régularité, recherche, progression, assiduité, initiatives ;
- la pertinence des réponses lors de l'entretien.

Attention à ne pas copier sans discrimination du code sur Internet. Tout emprunt de code devra être signalé. La présence de code non authentique et qui dénaturerait le projet sera sévèrement sanctionnée, vous risquez, au mieux, l'annulation de votre projet.

Afin de faciliter le suivi à distance, il est plus que souhaitable que tout votre code soit *systématiquement* disponible sur Github. M'envoyer le lien par mail. Il n'y a qu'un seul sujet pour la session 2.

Projet Jeu Flood-it

Vous devez réaliser un **clone** du jeu Flood-it comme il existait **originellement**, voir l'application Android **Floodit** qui restera le modèle graphique à répliquer. La nuance est que le choix des couleurs se fera **sur le plateau** de jeu et non pas dans une palette de couleurs séparée, comme dans cette **version en ligne**. L'application doit être pleinement jouable, sans bug et plaisante à utiliser.

A la base, le plateau est de taille 14x14, il y a 6 couleurs et le remplissage doit se faire en au plus 22 coups (et pas 25). Il doit y avoir un compteur de coups et un bouton pour relancer le jeu.

La difficulté algorithmique du jeu est l'algorithme de propagation de couleur. Cet algorithme devra être particulièrement soigné et bien maîtrisé. Il est attendu qu'il y ait une stricte séparation entre le modèle et la vue.

Cette partie est le strict minimum attendu et sera notée **au maximum** sur 75% de la note, bonus inclus.

Il est attendu, en complément, que soit réalisée une ou plusieurs IA minimalistes (des heuristiques) avec **animation** de résolution. Vous pouvez consulter ce **blog**.

Le jeu pourra contenir plusieurs bonus :

- menus déroulants (ou équivalent) pour le nombre de couleurs, la taille du plateau, etc
- bouton de backtrack
- chronomètre
- comparaisons d'IA sur des plateaux de très grande taille
- annonce au joueur d'une défaite certaine.

[Créé le mardi 18 mai 2021]