

I1 – Projet Informatique (2023/2024)

Ce projet est **noté sur 20** comme le DS. Il sera réalisé par **8 équipes (max.) de 3 ou 4 étudiants**.

Sujet

Écrire une application en python qui permet de simuler plusieurs animaux virtuels de type « Tamagotchi ». Le joueur doit s'occuper de 5 animaux et les maintenir en forme le plus longtemps possible. L'état de chaque animal est défini par plusieurs variables qui évoluent en fonction du temps et aussi en fonction des actions réalisées par le joueur.

Les informations d'état du tamagotchi sont les suivantes : **faim, santé, ennui**. Elles sont initialisées à 200 au début de la simulation. **Pour les équipes de 4 étudiants, il faudra aussi gérer la fatigue.**

La journée d'un tamagotchi dure 3 minutes avec une période de sommeil allant de 30 à 60 secondes (valeur aléatoire) :

- Chaque seconde de sommeil fait gagner 1 point de santé, 1 point d'ennui et **1 point de fatigue**.
- Chaque seconde d'éveil fait perdre 5 points de faim et 3 points d'ennui.
- Si l'ennui d'un des tamagotchis passe à zéro, il commence à se battre avec les autres et, tant qu'il est à zéro, tous les animaux perdent 5 points de santé par seconde.

Pour s'occuper des animaux, le joueur peut réaliser les actions suivantes :

- Donner une croquette à manger à un tamagotchi pour lui faire gagner 50 points de faim (on commence chaque journée avec 50 croquettes)
- Jouer avec un tamagotchi pour lui faire gagner 50 points d'ennui. **Par contre, le tamagotchi perd aussi 50 points de fatigue.**

Si la faim ou la santé **ou la fatigue** d'un seul des animaux passe à zéro, la partie est perdue.

Fonctionnement :

1. Le script affiche l'état des 5 tamagotchi puis attend une commande
2. Le script exécute la commande spécifiée (« manger » ou « jouer »)
3. ... le script recommence en ligne 1.

Attention : si la commande concerne un tamagotchi en sommeil, elle est annulée

Contrainte : les données d'état d'un tamagotchi devront être stockées dans un **dictionnaire**. Il y aura donc 5 dictionnaires ; ces 5 dictionnaires devront être stockés dans une **liste**.

Une partie de la note **(5 points)** sera consacrée aux **améliorations apportées au jeu**. Par exemple :

- Ajout d'une interface graphique (tkinter, web, pygame / pgZero, ...)
- Ajout de la possibilité de sauvegarder et restaurer la partie en cours
- Amélioration des règles du jeu
- Ajout de données d'état (soif, age (enfant/adulte), ...)
- Ajout d'actions (boire, nettoyer la cage, construire un nid, ...)
- ...
- **Pour les équipes de 3 étudiants : ajout de l'état de fatigue**

Une autre partie **(3 points)** sera consacrée à la **qualité du code** :

- Pertinence des noms des variables et des fonctions
- Intérêt des commentaires
- Découpage du projet en plusieurs fichiers « .py ».

Déroulement du projet

Le projet se déroule sur les 6 dernières séances du module (*une septième séance à part sera consacrée aux révisions pour le DS n°2*).

Séance 1 – Démarrage du projet

Cette première séance sera consacrée d'abord à la présentation du projet et, ensuite, à l'analyse du projet par les étudiants ainsi qu'à la constitution des équipes.

Il faudra constituer **8 équipes (max.) de 3 ou 4 étudiants**. **Un document Excel est partagé sur Moodle pour enregistrer ces équipes.**

La liste des équipes devra être finalisée **avant le début de la séance 2**.

Séance 2 – Spécifications du projet **(4 points – notation individuelle)**

Cette séance sera consacrée à l'élaboration et à la rédaction du document de spécification. Chaque étudiant devra rédiger un chapitre différent **(en indiquant son nom à côté du titre du chapitre)** du document de spécifications :

- **Fonctionnalités** – A partir de l'analyse du sujet, fournir la liste des actions (correspond généralement à une fonction « def ») à coder en Python pour que le jeu fonctionne
- **Structures de données** – Fournir aussi la liste des données importantes (cela peut être des listes, des dictionnaires, des classes ou même de simples variables)
- **Découpage en lots** – Découper le projet en modules (ou lots) répartis entre les membres de l'équipe.
- **(Uniquement pour les équipes de 4 étudiants) Propositions – Décrire des améliorations possibles du jeu (juste des idées, sans obligation de les implémenter)**

Ce document (~2 pages) devra être déposé au format PDF sur Moodle **avant la séance 3**.
(Bien indiquer le nom de l'étudiant à côté du chapitre qu'il a rédigé)

Séances 3 et 4 – Implémentation du projet (4 points)

Ces séances (et les éventuelles séances en autonomie) seront consacrées au codage du projet

Séance 5 – Test/Correction du jeu et préparation de la soutenance

Si ce n'est pas déjà fait, il va falloir intégrer les lots, développés précédemment, en un seul programme pour finaliser le jeu complet. Il faudra aussi tester le jeu final pour valider son fonctionnement. Enfin, le contenu et le déroulement de la soutenance sont à préparer.

Le code Python du projet devra être déposés sur Moodle **avant le jour de la soutenance.**

Séance 6 – Soutenance de fin de projet (4 points)

Chaque équipe disposera de 15 minutes pour réaliser sa soutenance :

- 10 minutes de présentation
- 5 minutes de questions/réponses + remarques de l'enseignant (feedback)

Structure de la présentation à respecter :

- Introduction avec noms des membres de l'équipe
- Spécifications fonctionnelles rapides avec surtout les améliorations apportées au projet
- Spécifications techniques : quelles structures (listes, ...) et dans quels buts ?
- Gestion du projet : façon de communiquer et de partager les codes
- Statut du projet/couverture fonctionnelle : qu'est-ce qui n'a pas pu être réalisé ?
- Démo rapide (ou vidéo) montrant ce qui fonctionne dans le projet
- Conclusion : que vous a apporté ce projet (en informatique et sur le travail en équipe) ?
- **Ne présentez pas de code Python...**

Vous pouvez créer votre présentation avec n'importe quel outil (Powerpoint, Libre Office, ...) mais il faudrait aussi en exporter une copie au format PDF pour être sûr que cela fonctionne sur n'importe quel système. Ayez une version de secours de votre présentation sur une clé USB au cas où votre PC aurait un problème ou s'il n'y a pas de réseau ...