

Enseignant(s)

**RAYNAL Benjamin**

Email(s)

[ben.raynal@gmail.com](mailto:ben.raynal@gmail.com)

## Snake Python IW2

### 1 Matières, formations et groupes

Matière liée au projet :

Formations : -

Nombre d'étudiant  
par groupe : **2 à 3**

Règles de constitution des groupes: **Libre**

Charge de travail  
estimée par étudiant : **10,00 h**

### 2 Sujet(s) du projet

Type de sujet : **Imposé**

## Snake/Tron

### Sujet :

Le but de ce projet est de réaliser un jeu vidéo en python de type snake/tron.

### Règles du snake :

Sur une grille rectangulaire, composée de cases, vous dirigez un serpent qui doit manger des pommes.

Chaque pomme fait grandir le serpent.

Le serpent avance tout seul, à rythme régulier (une case par seconde, par exemple).

Le serpent ne doit pas entrer en collision, ni avec un mur éventuel, ni avec lui-même.

La partie se termine quand une collision a lieu, et le score est alors le nombre de pommes mangées.

### Règle de Tron :

Pareil que pour le snake, mais plusieurs joueurs se partagent la même grille.

Une collision élimine le joueur

La partie se termine quand il ne reste qu'un seul joueur, qui est alors le gagnant

### Variations possibles:

Il est possible de faire différentes variations du jeu, combinables entre elles.

On peut recenser des modifications de la grille de jeu :

- grille torique : les bords de la grille ne sont plus considérés comme des murs, mais font apparaître de l'autre côté. Si on touche le mur gauche, on arrive sur le mur de droite, etc...

- trou de ver : des paires de cases spéciales dans la grille qui font que passer sur l'une, nous fait réapparaître sur l'autre

- murs aléatoires : des murs sont ajoutés aléatoirement à la grille au début de la partie. Il faut faire attention que l'ensemble des cases vides restent accessibles

- murs aléatoires dynamiques : pareil qu'au dessus, mais des murs s'ajoutent ou se retirent en cours de partie, sur des cases vides, en étant indiqués quelques tours à l'avance

- réduction : périodiquement, la taille de la grille diminue

On peut également définir de nouveaux types de bonus/malus à ramasser

- mauvaise pomme : une pomme qui fait perdre de la taille, et des points

- super pomme : une pomme qui fait grandir de plus d'une case, et donne plus de points

- pomme fantôme : permet à un joueur de passer à travers lui-même pendant un certain nombre de tours

- pomme de feu : les cases autour du serpent sont létales pour les serpents adverses

Toutes ces options peuvent bien entendu être combinées, et sont configurables.

### Consignes:

Il vous est demandé de réaliser ce jeu avec PyGame

Pour obtenir la moyenne, il faut au minimum avoir un jeu de snake fonctionnel avec une interface graphique et un tableau des scores persistant (qui est préservé entre plusieurs utilisations)

Pour obtenir 16, il faut réaliser un Tron en local (les contrôles des différents joueurs sont sur le même clavier), avec toutes les variations indiquées précédemment de disponible, activables et configurables via un menu.

Pour obtenir plus, il faudra faire plus (du réseau par exemple, ou plus d'options)

## 3 Détails du projet

### Objectif du projet (à la fin du projet les étudiants sauront réaliser un...)

Mettre ses connaissances du python en application

## Descriptif détaillé

Ouvrages de référence (livres, articles, revues, sites web...)

Outils informatiques à installer

pygame

## 4 Livrables et étapes de suivi

1

Rendu final

rendu final

dimanche  
06/02/2022  
23h59

## 5 Soutenance

Durée de présentation  
par groupe :

**10 min**

Audience : **A huis clos**

Type de présentation :

**Démonstration**

Précisions :