

## Travail pratique #1

### But du travail pratique

Le but de ce travail pratique est de vous familiariser avec

- La modélisation et la simulation sur des systèmes à faible complexité
- La simulation statique et stochastique en utilisant l'approche de Monte Carlo
- La génération de nombre (pseudo) aléatoires
- L'utilisation d'un logiciel de simulation

### Partie 1 - Introduction

Le nombre de touristes visitant le casino de Donald Trump à Las Vegas a diminué récemment (un peu comme sa popularité politique). Celui-ci vous a demandé d'aider à évaluer certains nouveaux jeux de cartes qui pourraient aider à augmenter le trafic touristique à ses tables de jeu.

Comme la plupart de ses touristes sont des Américains, il envisage des jeux de cartes très simples, avec peu de cartes, des règles simples et des bourses bien définies. Ce sont des jeux de cartes à un joueur qui coûtent 1 \$ pour jouer à chaque fois.

Votre tâche consiste à déterminer lequel de ces jeux de cartes est le plus attrayant (c.-à-d. possède la meilleure probabilité de gagner) ou le moins attrayant (c.-à-d. possède la plus faible probabilité de gagner) pour les joueurs.

Voici les nouveaux jeux de cartes considérés :

1. **Jeu 1** : le joueur pige une carte. Si celle-ci est un as, alors le joueur gagne 10\$. Autrement, le joueur perd.
2. **Jeu 2** : Le joueur pige deux cartes. Si les deux cartes sont exactement identiques, alors le joueur gagne 50 \$. Sinon, le joueur perd. (Avec remise)
3. **Jeu 3** : Le joueur pige deux cartes. Si la deuxième carte tirée a un rang strictement supérieur à la première carte tirée, alors le joueur gagne 2 \$. Sinon, le joueur perd. (Sans remise)
4. **Jeu 4** : Le joueur pige trois cartes. Si au moins une carte est de cœur, alors le joueur gagne k\$, où k est le nombre de cartes tirées qui sont des cœurs. Sinon, le joueur perd. (Sans remise)
5. **Jeu 5** : Le joueur pige cinq cartes. S'il existe un sous-ensemble de ces cartes qui fait une série de trois valeurs de cartes consécutives ou plus (par exemple, 5-6-7 ou 10-J-Q), alors le joueur gagne 5 \$. Sinon, le joueur perd. (Sans remise)

## **Partie 1 - Exigences**

Dans cette tâche, vous allez implémenter une simulation de type Monte Carlo (en C, C++, Java ou Python) pour évaluer et comparer les cinq jeux de cartes différents. Pour chaque jeu, vous simulerez 1 000 000 de parties du jeu afin d'estimer la probabilité de gagner.

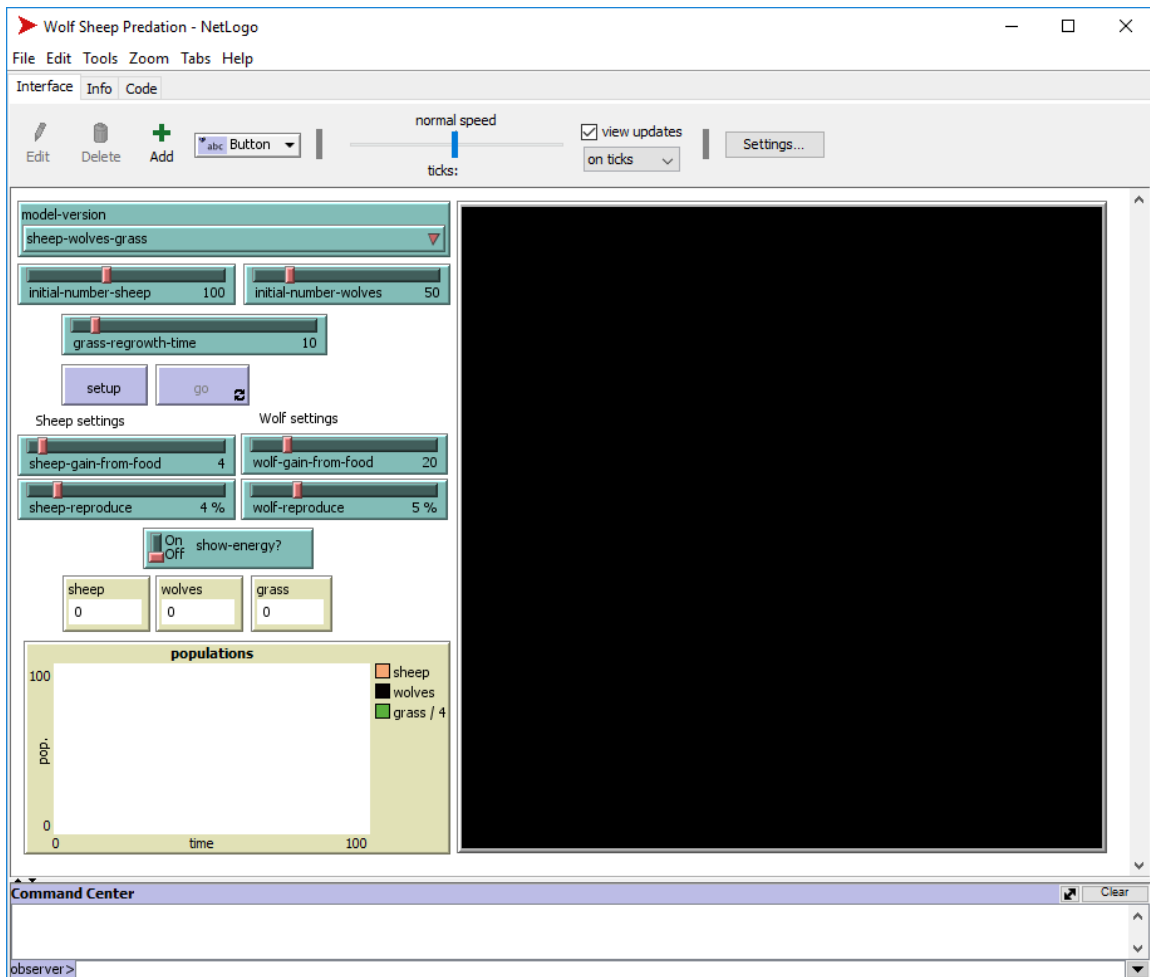
Vous devrez également concevoir et implémenter le code pour modéliser un croupier (dealer) qui travaille avec un jeu de cartes bien mélangé. Avec ce descriptif vous trouverez un exemple de code représentant un croupier, pour lequel la probabilité de recevoir une carte donnée (par exemple, le 5 de cœur) est toujours constante, quelles que soient les cartes précédemment pigées. N'hésitez pas à utiliser ce modèle comme point de départ, mais vous aurez normalement besoin d'un modèle de croupier plus réaliste afin de compléter vos expériences Monte Carlo.

## **Partie 2 – Introduction**

Nous verrons dans la théorie du cours 4 le modèle Prédateur-Proie qui représente une population de moutons et de loups. Vous devrez télécharger un logiciel de simulation afin de répondre à la question présentée ci-dessous.

## Partie 2 - Exigences

1. Tout d'abord, vous devrez télécharger et installer le logiciel de simulation NetLogo disponible gratuitement à l'adresse suivante : <http://ccl.northwestern.edu/netlogo/>. Alternativement, vous pouvez utiliser le simulateur NetLogo version Web.
2. Ensuite, vous devrez ouvrir le modèle « Wolf Sheep Predation » (File > Models library et ouvrir la catégorie « Biology »). Sélectionnez la version « sheep-wolves-grass » du modèle :



3. Utilisez le modèle afin de déterminer pourquoi la population de loups se retrouve rapidement à 0 lorsque le paramètre « grass-regrowth-time » est à 10. **Laissez les autres paramètres aux valeurs par défaut.**
4. Faites une capture d'écran de votre simulation.

## Document à remettre

Vous devrez remettre les documents suivants :

- **Un fichier PDF contenant les parties 1 et 2:**
  - Les résultats obtenus lors de vos simulations ainsi qu'une analyse de ceux-ci
  - Le manuel d'utilisation de vos programmes (partie 1).
- **Vos fichiers C/C++/Java/Python** créés afin de réaliser vos simulations (partie 1).

## Barème de correction (15% total)

### Partie 1 (10% de la note)

- (4%) Conception et mise en œuvre d'un simulateur Monte Carlo approprié (c.-à-d., boucle itérative principale, génération de nombres aléatoires, sortie statistique, etc.)
- (2%) Mise en œuvre appropriée d'un croupier à un seul jeu de cartes (c.-à-d., fonctionnalité, exactitude, efficacité)
- (2%) Implémentation correcte des jeux de cartes indiqués ci-dessus
- (2%) Tableau des résultats de simulation estimant la probabilité de gagner chaque partie dans les différents scénarios

### Partie 2 (5% de la note)

- (5%) Analyse des résultats et captures d'écran de votre simulation

## Date de remise

Le lundi 21 février 2022 via le Moodle du cours.