

L'objectif de ce projet est d'illustrer l'utilisation des concepts et mécanismes abordés en cours de Systèmes & Réseaux au travers du développement du jeu de carte *6 qui prend* sur ordinateur en permettant aux utilisateurs de jouer en ligne à travers un terminal avec d'autres joueurs et/ou processus robots.

6 qui prend est un jeu de carte de type "défausse" créé par Wolfgang Kramer en 1994. En voici une description issue du site Gigamagic : "dans ce jeu culte, il vous faudra placer vos cartes dans les différentes rangées mais sans jamais poser la 6ème! Les choix des cartes étant secret et simultané, ils vous faudra être un poil calculateur pour ne pas finir avec le plus gros troupeau. Alors gare aux vacheries!"

Principe du jeu : Le but de ce jeu de cartes est d'avoir le moins de têtes de bœuf à la fin de la partie. Les cartes ont 2 valeurs, une valeur numérique (de 1 à 104) et une valeur de 1 à 7 représentée par des "têtes de bœufs".

Chaque manche débute avec 4 cartes tirées au hasard qui formeront 4 rangées. Lors de chaque manche, chaque joueur reçoit 10 cartes au hasard, il y aura donc 10 tours par manche. Les joueurs choisissent ensuite une carte parmi les dix et la posent face cachée. Elles seront placées plus tard dans une des 4 rangée de la plus petite à la plus grande.

Au bout d'un moment certaines rangées contiendront 5 cartes et lorsque l'un des joueurs pose sa carte en sixième place il prend les 5 cartes et pose la sienne à la première place de la rangée. Lorsqu'un joueur pose une carte plus petite que celles déjà présentes sur la table il ramasse la rangée de son choix (normalement celle qui contient le moins de tête de bœuf).

A la fin de chaque manche chacun augmente son score avec le nombre de têtes de bœufs qu'il a ramassé. Le joueur qui a atteint 66 têtes de bœufs (ou autre valeur à définir, ou nombre fixé de manches) a perdu la partie.

Afin de mieux comprendre les règles du jeu, vous pouvez regarder des vidéos de description en ligne : [ici](#) ou [là](#).

Fonctionnalités à développer

Le système comprendra trois types de processus :

Gestionnaire du jeu : accueil des joueurs et des manches, distribution des cartes, dépose des cartes jouées sur les rangées; information aux joueurs; compte des têtes de bœufs par joueur; etc

Joueur Humain : permet à l'utilisateur de jouer ses cartes au moment voulu : affiche les début et fin de parties, transfère au gestionnaire les ordres de jeu de l'utilisateur et affiche les cartes jouées par les autres et le résultat

Joueur Robot : participe au jeu comme un joueur humain en tentant de jouer les cartes reçues pour obtenir le moins de têtes de bœufs possibles.

Votre système devra au minimum permettre à des joueurs humains et robots de jouer sur la même machine, dans une version avec où chaque carte vaut une tête de bœuf. Puis des fonctionnalités additionnelles peuvent être développées, parmi lesquelles :

- nombre de têtes de bœufs variable par carte ;
- jeu avec utilisateurs en réseau ;
- joueurs robots plus intelligents ;
- sauvegarde des parties dans fichiers de log et production d'un document pdf montrant le déroulement de la partie ;
- génération de statistiques (par exemple points obtenus et parties gagnées/perdus par utilisateur, points moyens par parties, etc) ;
- comparaison statistique de votre stratégie de joueur robot avec la stratégie de jouer les cartes au hasard.

Modalités

- Le projet est à faire en binôme (demander explicitement l'accord du chargé de TP en cas d'autre configuration). La participation de chacun des membres du binôme devra apparaître sur les rapports et lors de la présentation.
- Le projet doit être réalisé en utilisant principalement les langages de scripts awk/shell et/ou en C.
- Un rapport préliminaire succinct (1 page, pdf uniquement) doit être déposé sur plubel le 13 novembre au plus tard. Il contiendra, outre le nom du binôme, un premier aperçu de la structure générale de l'application et de la façon de réaliser les fonctionnalités demandées (choix des mécanismes de communication, gestion du temps, etc). La pertinence de ce rapport interviendra dans la note de rapport.
- Une démonstration du projet sera faite en salle machine (ou sur portable, pour ceux qui préfèrent) la semaine du 12 décembre créneaux définis ultérieurement).
- Le rapport final doit être déposé sur plubel au plus tard au moment de la démo.
- l'évaluation portera sur
 - le projet (/10) : fonctions implantées, pertinence des outils et langages utilisés, propreté du code,
 - la démo (/5) : respect du temps, partage de parole, préparation (jeux de tests),
 - le pré-rapport/rapport (/5) : structure, contenu, recul sur le sujet.