

## Sujets de projet

Vous devez choisir l'un des sujets de projet proposés ci-après, et uniquement l'un d'entre eux.  
Pour tous les sujets suivants, on vous demande d'écrire un programme PROLOG permettant de résoudre aussi efficacement que possible le problème décrit.

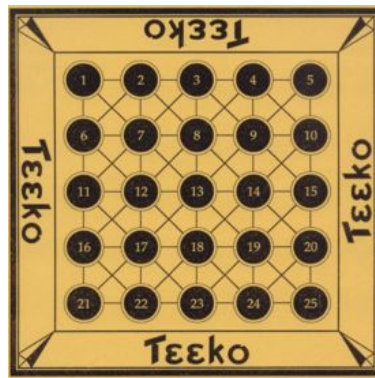
Merci de respecter à la lettre les consignes suivantes :

- Chaque projet devra être réalisé par groupe de **deux personnes minimum** ou **trois personnes maximum**. **Il ne sera accepté aucun projet réalisé par une seule personne ou par quatre personnes ou plus.**
- **La constitution des groupes ainsi que le choix du sujet devront être communiqués par mail à [fabrice.lauri@utbm.fr](mailto:fabrice.lauri@utbm.fr) pour le 22 Octobre 2007 au plus tard.**
- Les projets devront être rendus dans la **semaine du 7 Janvier**, à votre **enseignant de TP pendant la séance**, à la fois **par mail et par exemplaire papier**. Pour chaque jour de retard, **deux points** seront déduits de la note de projet.
- L'évaluation du projet s'appuiera sur :
  - la qualité globale du rapport (1/3 de la note)
  - la qualité des code sources PROLOG (1/3 de la note)
  - la qualité de la présentation (1/3 de la note) d'une quinzaine de minutes qui se déroulera la **semaine du 14 Janvier** pendant la séance de TP.
- Les éléments suivants seront examinés lors de l'évaluation du projet :
  - les phases de spécification et d'analyse du problème,
  - le mécanisme mis en oeuvre pour déterminer une solution,
  - la qualité du code source,
  - la qualité des solutions trouvées,
  - le nombre de situations traitées.
- Le rapport, de **15 pages au maximum**, devra comporter au moins :
  - un rappel de l'énoncé du problème,
  - votre spécification (formalisation) du problème,
  - l'analyse du problème,
  - la méthode proposée, avec en annexe le **listing du programme commenté**,
  - la description d'une ou de plusieurs situations traitées par votre programme,
  - les résultats obtenus par votre programme sur ces situations,
  - les difficultés éventuellement rencontrées, les améliorations possibles (méthodes de résolution) et les perspectives d'ouverture possibles du sujet traité.

Certains des problèmes ci-après semblent nécessiter un apport de connaissances immense pour pouvoir être traités correctement. Apporter donc le plus grand soin dans la phase de spécification de votre programme (ce que vous voulez qu'il réalise), en définissant le plus précisément possible son domaine d'étude. Un conseil : consacrez au minimum deux à trois semaines pour cette phase, la qualité de votre projet en dépend...

## Sujet #1

### Jeu à deux adversaires : Teeko



**Nombre de joueurs :**

2

**Matériel :**

- plateau carré dont chaque côté est de longueur 4
- 4 pions noirs et 4 pions blancs

**But du jeu :**

Placer ses pions puis les déplacer pour former un alignement (horizontal, vertical ou en diagonal) ou un carré de 4 pions.

**Description du jeu :**

Le plateau est vide en début de partie. Tour à tour, les joueurs posent un de leurs pions sur une intersection libre (emplacements de 1 à 25 sur la figure ci-dessus). Une intersection est libre si elle ne contient pas déjà un pion. A la fin de cette phase de pose, si aucun joueur n'a obtenu de configuration gagnante (alignement ou carré de 4 pions), les joueurs déplacent à tour de rôle l'un de leurs pions. Un pion déplacé ne peut l'être que sur un emplacement libre adjacent. Dès qu'un joueur réalise une configuration gagnante, la partie s'arrête.

**Objectif du projet :**

Programmer l'IA d'un joueur ou des deux.

## Sujet #2

### Dialogue homme-machine : Le Shmilblic



#### Description du jeu :

Lancé en septembre 1969, le *Schmilblic* était animé par Guy Lux, qui présentait une photo en très gros plan d'un objet. Les candidats lui posaient des questions auxquelles il pouvait répondre oui ou non. Si la réponse était positive, de l'argent était ajouté dans la cagnotte et les candidats pouvaient alors formuler une proposition. Si la proposition correspondait à l'objet caché, le candidat remportait la cagnotte. La fameuse parodie de Coluche a grandement participé à la popularité de ce jeu.

Pour ce projet, on supposera qu'il n'y a qu'un seul joueur et que ce joueur doit découvrir l'**objet**, la **personne** ou l'**animal**, dissimulé par l'ordinateur sous le mystérieux nom de *Schmilblic*.

Le joueur pose une série de questions dans le but de faire avancer le *Schmilblic*, c'est-à-dire de progresser vers la solution. Le joueur doit trouver la solution en un nombre fixé N de questions. Les questions ne peuvent appeler que les réponses OUI ou NON.

Lorsque c'est NON, le joueur peut reposer une question. Lorsque c'est OUI, le joueur peut reposer une autre question ou proposer une réponse. Si la réponse est juste, il a gagné.

#### Objectif du projet :

Le joueur devra poser ses questions dans un langage proche du français. Une fois la question posée, l'ordinateur devra reconnaître sur quoi porte la question et répondre par OUI ou par NON en fonction du *Schmilblic* qu'il a choisi.

### Sujet #3

## Mise en oeuvre d'une IA dans un jeu vidéo : Delirium 2



#### Description du jeu :

Le joueur incarne un minier (en rose en haut à gauche dans l'image ci-dessus) qui évolue en creusant dans des sous-terrains composés de diamants et de pierres et peuplées de monstres, entre autres.

Le but du jeu est de récolter un certain nombre de diamants, pour pouvoir s'engouffrer vers l'accès qui mène au prochain sous-terrain, cet accès restant bloqué tant que ce nombre n'est pas atteint.

Lorsque le minier se déplace, il laisse derrière lui une case vide, si bien que diamant et pierre peuvent tomber en chaîne. Le minier meurt s'il reçoit un diamant ou une pierre sur la tête.

#### Objectif du projet :

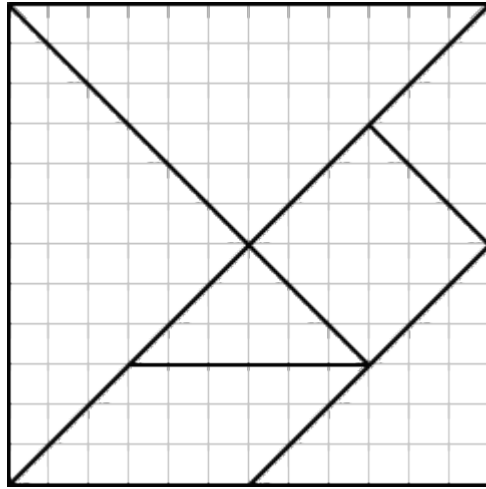
Programmer l'IA du minier afin qu'il puisse récolter tous les diamants nécessaires à l'accès au sous-terrain suivant.

Ce jeu a été écrit en C++ et OpenGL et est opérationnel : plusieurs sous-terrains sont disponibles, les monstres se déplacent et les pierres ainsi que les diamants chutent.

Il est accessible dans le répertoire Delirium2. Pour pouvoir être exécuté, OpenGL doit être installé sur votre ordinateur. Consultez le fichier ReadMe.txt pour savoir quelles touches utiliser dans le jeu...

## Sujet #4

### Casse-tête : Le tangram



#### Description du jeu :

Le Tangram est un jeu très ancien, d'origine chinoise. C'est un jeu de réflexion comparable à un puzzle. Les 7 sept pièces de base du puzzle sont obtenues très simplement à partir du découpage d'un carré : deux petits triangles rectangles, un triangle rectangle moyen, deux grands triangles rectangles, un carré et un parallélogramme, la seule pièce sans axe de symétrie, donc le plus difficile à manier...

On réalise des silhouettes avec les 7 formes indiquées dans le dessin ci-dessus. Ce peuvent être des personnages, des lettres, des chiffres des formes géométriques...

#### Objectif du projet :

L'utilisateur décrit le contour d'une forme et le programme doit reconstituer cette forme à l'aide des pièces du Tangram.