

## 1 Travail demandé

Le travail demandé comporte une implantation en Langage orienté objet. Un rapport court (1 à 4 pages) devra accompagner le projet au format papier.

### Ce rapport devra

- expliquer vos choix de modélisation et lever les ambiguïtés du sujet ;
- mettre des diagrammes d'héritages et de classes/prototypes ;
- poser les algorithmes principaux sous forme synthétique (pas le code !);
- présenter la répartition du travail ;

### Implantation

- Votre programme devra utiliser un langage orienté objet : Java, C++, Eiffel, JavaScript, Lisaac, etc.
- L'interface utilisateur est graphique.
- Une petite explication de l'utilisation de votre programme est aussi nécessaire.

## 2 Réalisation du projet

La remise et l'évaluation du projet se feront lors d'une soutenance de projet en fin de semestre (à l'heure actuelle, nous n'avons pas encore fixé de date précise ou salle). Les sources seront envoyées, la même semaine, par courrier électronique à votre responsable de TP :

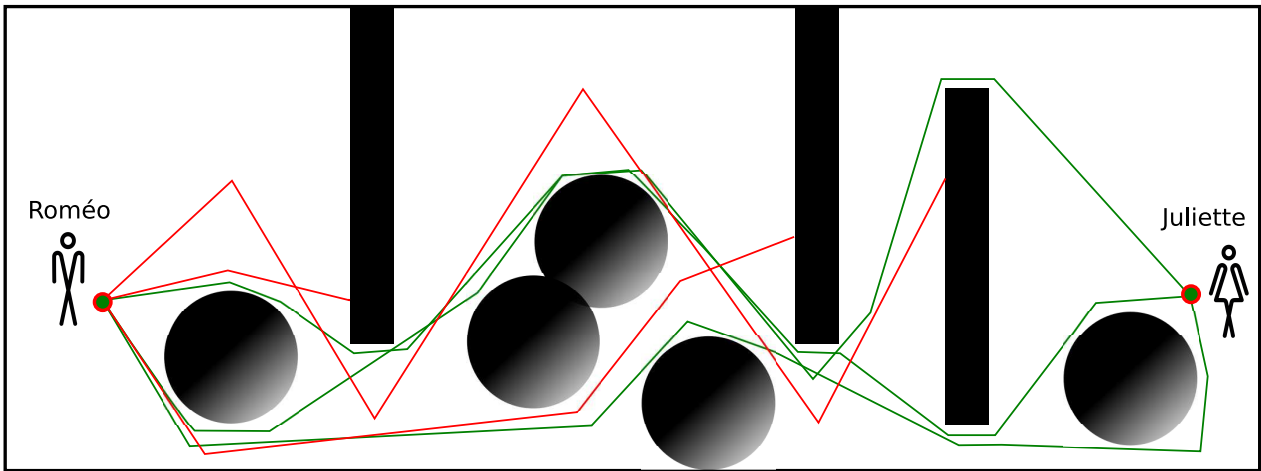
`benoit.sonntag@lisaac.org`

- La réalisation de ce projet devra se faire impérativement par groupe de deux (ou seul) pour que vous vous répartissiez le travail.
- Comme tout cahier des charges, celui-ci ne peut être exhaustif. En cas d'ambiguïté, préciser votre interprétation personnelle, et éventuellement les questions à poser à votre interlocuteur (responsable de projet, futurs utilisateurs, etc.). Toute solution cohérente, justifiée et non contradictoire avec le cahier des charges sera acceptée.

## 3 Sujet

L'objectif est de trouver un chemin entre deux points (dans un espace à deux dimensions) tout en évitant des obstacles aléatoires. Vous avez le choix de l'algorithme pour la réalisation.

Ici, nous avons un point de départ (*Roméo*) et un point d'arrivée (*Juliette*). Les obstacles sont des formes simples (disque, rectangle). Les points source/destination sont toujours à la même position. Ensuite, vous positionnez aléatoirement des obstacles sur cet espace fermé. La taille et la concentration d'obstacles sont des paramètres de votre programme. Le monde étant installé, Roméo doit chercher un chemin pour rejoindre Juliette.



*Rmq.* : Les algorithmes génétiques apportent de bon résultat pour ce type de problème.

#### 4 Idées d'extension du sujet

- Évidemment, par construction, il est possible qu'il n'y ait pas de chemin pour atteindre Juliette. Vous pouvez améliorer la construction de votre monde en créant dès le départ un chemin aléatoire entre Roméo et Juliette, puis poser des obstacles en dehors de ce chemin. Ainsi, nous garantissons une solution (Roméo ne doit pas la connaître à l'avance et trouver de son propre chef le chemin qui mène à Juliette).
  - Trouver un chemin, c'est bien. Trouver le meilleur chemin c'est mieux !
  - Mettre en place du parallélisme pour profiter des différents cœurs de votre processeur.
  - Imposer un temps limite de recherche à votre programme.
- Toute extension sera la bienvenue et sera prise en compte.

Good luck !