

## IA54

## Final A2014

Tous documents autorisés,  
Tous dispositifs électroniques interdits  
Le barème est donné à titre indicatif (±1)

**1 copie par exercice****Exercice 1 (8 points) – Système intelligent de transport à la demande.**

Considérons RAPIDO, l'entreprise de transport collectif à la demande de la région Belfortaine. Nous souhaitons développer à l'aide du paradigme multiagent une application informatique destinée à planifier les tournées journalières des  $n$  véhicules de l'entreprise RAPIDO en fonction des demandes des utilisateurs.

L'organisation des tournées s'appuie sur quatre principaux modules :

- **La collecte des demandes clients** : Un client effectue depuis son téléphone portable une demande de transport sur l'application en précisant le jour, le lieu et l'heure de départ souhaitée ainsi que le lieu et l'heure d'arrivée souhaitée.
- **La planification des tournées** : A partir des demandes collectées, le système planifie les tournées à venir ou peut éventuellement modifier une tournée en cours lorsque cela est pertinent.
- **La réalisation de chaque tournée pour chaque chauffeur** : Une fois la tournée définie ou modifiée, les informations de son parcours (lieux et horaires) sont envoyées au PDA de chaque chauffeur.
- **La régulation** : Le régulateur dispose d'une application lui permettant de visualiser en temps réel la localisation et le parcours effectué pour chaque triplet (véhicule, chauffeur, clients).

On considère que nous disposons déjà d'un module informatique *TOURMANAGER* capable de faire la planification ou la re-planification des tournées en fonction des demandes clients, et de la localisation des chauffeurs.

La modélisation proposée du système devra au moins intégrer les agents associés aux clients, aux chauffeurs et aux éléments nécessaires au bon fonctionnement de l'application informatique de RAPIDO.

**Question 1 (4 points)** : Décrire à l'aide du langage CRIO le modèle organisationnel de l'application RAPIDO : les différentes organisations, les rôles qui les composent et les capacités associées.

**Question 2 (4 points)** : Décrire un exemple d'instanciation de ce système présentant les différents agents logiciels nécessaires, les rôles qu'ils jouent au sein des différents groupes instanciant les organisations décrites à la question précédente.

## IA54

### Final A2014

Tous documents autorisés,  
Tous dispositifs électroniques interdits  
Le barème est donné à titre indicatif (±1)

### 1 copie par exercice

#### Exercice 2 (4 points) – Plate-forme SMA.

Un agent doit pouvoir migrer d'un micro-noyau vers un autre micro-noyau. Pour cela, l'agent doit disposer de suffisamment de ressources sur le noyau distant avant de pouvoir migrer. Il est par conséquent nécessaire que le micro-noyau de départ garantisse la disponibilité des ressources à l'agent. Pour cela, il doit communiquer avec l'autre micro-noyau pour obtenir la quantité de ressources disponibles. Nous considérerons dans cet exercice que la mémoire disponible.

La fonction demandant la migration d'un agent à un micro-noyau est (syntaxe SARL) :

```
def migrateTo(remoteKernel : Address, agent : Agent) : boolean
```

Le paramètre `remoteKernel` est l'adresse du noyau destination. Vous supposerez que l'agent connaît cette adresse. Le paramètre `agent` est la référence sur l'agent voulant migrer. La fonction `migrateTo` retourne une valeur booléenne indiquant si la migration est autorisée par le micro-noyau.

La fonction suivante permet à un micro-noyau d'obtenir la taille de mémoire nécessaire pour un agent (syntaxe SARL) :

```
def computeAgentSize(agent : Agent) : long
```

La fonction suivante permet à un micro-noyau d'obtenir la taille de la mémoire libre sur son calculateur (syntaxe SARL) :

```
def getSystemFreeMemory : long
```

**Question 1 (2 points)** : Écrivez un diagramme de classe décrivant les éléments constituant la plate-forme et uniquement nécessaires au processus de migration. Vous ignorerez les éléments de la plate-forme n'intervenant pas dans ce processus.

**Question 2 (2 points)** : Écrivez un diagramme de séquence décrivant le processus de migration d'un agent. Les entêtes des lignes de vie seront les éléments de votre diagramme de classe. Les messages décrits entre les lignes de vie seront les messages échangés ou les appels de fonctions existants entre les micro-noyaux et/ou l'agent voulant migrer.

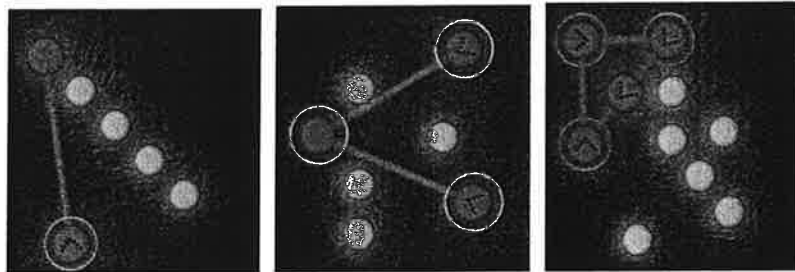
**1 copie par exercice**

**Exercice 3 (5 points) – Agents réactifs et résolution de problème.**

**Nodes** est un jeu sur plate-forme mobile de type puzzle-game. Le but de ce jeu est de transpercer tous les « nœuds » bleus pour les allumer en rouge. Pour cela, le joueur doit manœuvrer des rayons lasers pour atteindre l'objectif. Au fil de la progression, les « nœuds » à allumer se font de plus en plus nombreux, et doivent parfois être transpercés plusieurs fois par un laser pour compléter le niveau. Des éléments supplémentaires, tels que des miroirs, peuvent également faire une apparition. Dans cet exercice on se propose de trouver une solution réactive aux problèmes posés par le jeu. On se limitera aux premiers niveaux du jeu que font intervenir uniquement les éléments suivants :

- Nœuds bleus : ils doivent être traversés par le laser. Ils sont fixes dans l'environnement.
- Nœuds rouges simples : ils sont des puits (c'est le point d'arrivée du laser). Ils sont fixes dans l'environnement.
- Nœuds rouges avec un cercle supplémentaire : Ce sont des puits mobiles.
- Nœuds rouges avec une flèche : Ce sont des sources de laser. Le sens de la flèche indique le sens d'émission du laser. Ils sont fixes dans l'environnement.
- Nœuds rouges avec une flèche et cercle supplémentaire : Ce sont des sources mobiles.

L'ordre de passage dans les nœuds est fixe et défini au départ. Les nœuds mobiles se déplacent en rotation autour du nœud suivant dans le parcours du laser (l'orientation va toujours vers le nœud suivant). Il est également possible d'augmenter la distance entre deux nœuds. Les figures suivantes montrent quelques configurations possibles.



**Question 1 (2 points) :** On veut résoudre ce jeu en utilisant des agents réactifs. Quelles sont les méthodes possibles vues en cours permettant de résoudre ce problème. Listez juste les méthodes sans les détailler.

**Question 2 (3 points) :** Choisissez l'une de ces méthodes et instanciez là de façon à résoudre le problème. Vous définirez de façon détaillée, les agents, leurs comportements, les interactions, l'environnement... N'hésitez pas à utiliser des diagrammes vus en cours.

## IA54

### Final A2014

Tous documents autorisés,  
Tous dispositifs électroniques interdits  
Le barème est donné à titre indicatif ( $\pm 1$ )

### 1 copie par exercice

#### Exercice 4 (3 points) – Apprentissage par renforcement

Soit le processus décisionnel de Markov  $M = (S, A, T, R, \lambda)$  défini par :

- $S = \{ 1, 2, 3, 4 \}$  l'ensemble des états ;
- $A = \{ a1, a2 \}$  l'ensemble des actions ;
- $T = \{ (1,a1,2), (1,a2,3), (2,a1,1), (2,a2,4), (3,a1,1), (3,a2,2), (4,a1,4), (4,a2,4) \}$  l'ensemble des transitions entre états ;
- $R$  la fonction de récompense telle que :
  - $R(s,a) = 1$  si  $(s,a,4) \in T$  (autrement dit si l'application de l'action  $a$  dans l'état  $s$  mène à l'état 4),
  - $R(s,a) = 0$  sinon ;
- $\lambda = 0,9$

**Question 1 (1,5 points) :** Appliquez l'algorithme Q-iteration pour déterminer la matrices des Q-valeurs après 3 itérations.

**Question 2 (1 point) :** Donnez la politique  $\pi$  obtenue à partir de la matrice précédente des Q-valeurs.

**Question 3 (0,5 point) :** Cette politique est-elle optimale ? Justifiez votre réponse.