



**Corrigée type d'examen du Module Modélisation & simulation (2<sup>ème</sup> année master IDO)**

Nom : ..... Prénom : .....

**/20**

➤ **Questions de compréhension des cours** : mettez **des croix** sur les bonnes réponses

<b>1- La simulation permet de comprendre :</b>
- Les entrées et sorties d'un système
- Les composants et les relations qui les relient dans un système
- Le fonctionnement d'un système <input checked="" type="checkbox"/>
<b>2- La Simulation a pour but de :</b>
- Minimiser les couts de développement d'un système <input checked="" type="checkbox"/>
- Minimiser les erreurs de développement d'un système <input checked="" type="checkbox"/>
- Augmenter les performances d'un système <input checked="" type="checkbox"/>
<b>3- Un système à évènements discrets est :</b>
- Un système basé sur les évènements <input checked="" type="checkbox"/>
- Un système basé sur les processus
- Un système basé sur les actions
<b>4- La simulation est un outil d'aide à la décision très utilisé dans :</b>
- Les systèmes simples
- Les systèmes complexes <input checked="" type="checkbox"/>
- Les systèmes mixtes
<b>5- Un système multi agents (SMA) est un ensemble d'agents qui :</b>
- ont des rôles indépendants
- ont un objectif commun <input checked="" type="checkbox"/>
- ont des rôles relatifs
<b>6- Un agent peut-être implémenter sous forme de :</b>
- Un Objet
- Un Objet autonome
- Un processus autonome <input checked="" type="checkbox"/>
<b>7- Un SMA composé des agents cognitif peut contenir :</b>
- Un nombre important d'agents
- Un petit nombre d'agents <input checked="" type="checkbox"/>
- Un nombre moyen d'agents
<b>8- La simulation d'un système est basée sur :</b>
- L'élaboration d'un prototype descriptif <input checked="" type="checkbox"/>
- L'élaboration d'un programme descriptif
- L'élaboration d'un modèle descriptif <input checked="" type="checkbox"/>
<b>9- Le langage FIPA-ACL est un :</b>
- Langage de communication dans un SMA <input checked="" type="checkbox"/>
- Langage de négociation dans un SMA
- Langage de coopération dans un SMA
<b>10- Un programme dans la plateforme JADE est :</b>
- basé sur le langage JAVA <input checked="" type="checkbox"/>
- basé sur le langage FIPA-ACL
- basé sur le langage KQML
<b>11- La plateforme JADE contient :</b>
- Des groupes d'agents
- Des conteneurs d'agents <input checked="" type="checkbox"/>
- Des Objets
<b>12- Un modèle à évènement discrets représente</b>

- La partie statique d'un système
- La partie dynamique d'un système
- Les deux parties statique et dynamique X

### 13- Les systèmes multi agents sont inspirés de :

- L'intelligence artificielle
- L'intelligence artificielle distribuée X
- Systèmes experts

### 14- Pour simuler un système, il faut :

- Modéliser sa structure
- Modéliser son comportement
- Modéliser sa structure et son comportement X

➤ **Exercice** : Soit l'exemple de deux agents, citez par énumération les étapes d'exécution de ce code

```
public class AgentA extends Agent
{
protected void setup(){
private class attendreAgentB extends OneShotBehaviour
{
public void action() {
System.out.println("en attente de l' agent B");
block();
}
}
private class envoi extends OneShotBehaviour{
public void action() {
ACLMessage messageRecu = receive();
if (messageRecu.getContent().equals ("pret") )
{
ACLMessage message = new
ACLMessage(ACLMessage.INFORM) ;
message.addReceiver(messageRecu.getSender());
message.setContent("Je suis actif");
send(message); }
else
myAgent.doDelete();
}
} }
}
```

```
public class AgentB extends Agent {
boolean stop = false;
protected void setup(){
private class attendre extends OneShotBehaviour {
public void action() {
ACLMessage message = new
ACLMessage(ACLMessage.INFORM);
message.addReceiver(new AID("AgentA",
AID.ISLOCALNAME));
if(!stop) {
message.setContent("pret");
send(message);
block();
}else{
message.setContent("arret");
send(message);
}
} }
private class afficher extends OneShotBehaviour {
public void action() {
ACLMessage messageRecu = receive();
System.out.println("message reçu= "+
messageRecu.getContent()); stop = true;
}
}
}}
```

- 1- Un agent A attend un message d'un agent B, il affiche le message **en attente de l'agent B** et se bloque en attente du message
- 2- L'agent B crée un message **INFORM** avec le contenu **prêt**
- 3- L'agent B teste s'il n'est pas dans un état **stop** alors il envoie le message à l'agent A et se bloque
- 4- Si l'agent B est dans un état **stop** alors le message à envoyer prend le contenu **arrêt**
- 5- L'agent A reçoit un message de l'agent B et teste son contenu s'il est égal à **prêt**
- 6- Si oui alors L'agent A crée un message **INFORM** avec le contenu **je suis actif**
- 7- L'agent A envoie le message à l'agent B et se bloque
- 8- Si le message reçu est égal à **arrêt** alors l'agent A se détruit
- 9- L'agent B reçoit le message
- 10- L'agent B affiche le message **message reçu je suis actif**
- 11- L'agent B met son état à **stop**