

## Cours : Spécification et vérification formelles

Classe : 3<sup>ème</sup> années Ingénierie des Systèmes Intelligents

### TD Spécification formelle : Schémas

**Exercice 1** : Une compétition de football réunit plusieurs équipes. Pour participer, chaque équipe doit s'inscrire. Un numéro séquentiel lui est donné pour l'identifier pour la suite de la compétition. À une date fixée par l'organisation, l'inscription est close. La compétition démarre alors. Elle se déroule sous forme de tours éliminatoires, lors desquels les équipes s'affrontent deux à deux dans des matchs. L'équipe qui gagne son match est qualifiée pour le tour suivant. Toutes les équipes inscrites sont qualifiées pour le premier tour. Les matchs d'un tour sont tirés au sort. La première équipe tirée au sort reçoit l'équipe adverse. Si le nombre d'équipes est impair, la dernière équipe est qualifiée d'office pour le tour suivant. La compétition s'arrête lorsqu'elle ne reste qu'une équipe.

Lorsqu'un match d'un tour est joué, l'équipe gagnante est qualifiée et l'organisation mémorise sous forme de texte non précisée l'équipe gagnante, le score dans le temps réglementaire, le score après prolongation et le score après d'éventuels tirs aux buts. Lorsque tous les matchs d'un tour sont joués, l'organisation effectue le tirage au sort du tour suivant. On utilise les types suivants :

[EQUIPE, TEXTE]

ETAT\_COMP ::= inscription | tirage | tour | fin

1. Définir la fonction **tirage** qui prend aléatoirement un entier dans un ensemble d'entiers
2. Définir la fonction **classe** qui définit les partitions d'un ensemble E d'entiers en sous-ensembles de taille 2. Le cardinal de E est supposé pair.
3. Représenter le système sous forme d'un schéma Compétition en indiquant les types nécessaires
4. Donner l'état initial
5. Décrire les opérations suivantes sous forme de schémas:
  - a. **Inscrire** : inscrire une équipe. Elle prend en entrée une équipe et retourne le numéro attribuée à cette équipe
  - b. **CloreInscription** : close les inscriptions. Elle ne prend aucune entrée et ne retourne aucun résultat.
  - c. **Tirage** : Effectuer le tirage d'un nouveau tour. Elle ne prend aucune entrée et ne retourne aucun résultat.
  - d. **Résultat** : Rentrer le résultat d'un match. Elle prend en entrée les deux équipes qui ont joué ainsi que le gagnant du match et le résultat.

e. **TourSuivant** : Passer au tour suivant. Elle ne prend aucune entrée et ne retourne aucun résultat.

6. Énoncer les théorèmes de l'initialisation et des pré-conditions

**Exercice 2** : Un système de surveillance d'un supermarché sert à faire le suivi des files d'attente devant les différentes caisses. Nous modélisons les caisses par l'ensemble *Caisses*. Une caisse est spécifiée par le type de base *CAISSE*. Si une caisse est en service, le système doit enregistrer l'état de la file des personnes pour cette caisse. Nous faisons abstraction des détails pour la spécification d'une personne et la modélisons par le type de base *PERSONNE*. Une propriété que le système devra vérifier en tout temps est qu'une personne ne peut se trouver dans deux files différentes au même moment. Toute personne qui a terminé de remplir son chariot peut choisir n'importe quelle caisse en service. Toute personne qui n'est pas en tête de file, peut la quitter et joindre une autre file d'une autre caisse en service.

1. Spécifier l'état général du système sous forme d'un schéma qu'on nommera *Supermarché*.
2. Spécifier sous forme d'un schéma *PlusDeCaisse* les états du système où il n'y a plus de caisses qui pourront être en service.
3. Spécifier sous forme du schéma *CaissesPleines* les états du système où chaque caisse en service a plus que trois personnes en attente.
4. Écrire un prédicat qui stipule qu'il existe un état dans lequel toutes les caisses sont en service et sont pleines (plus que trois personnes en attente)
5. Spécifier l'état initial du système
6. Spécifier sous forme de schéma, l'opération *JoindreFilePartielle* qui décrit l'effet sur le système lorsqu'une personne décide d'aller payer et joindre une file d'attente. Cette opération prend en entrée, une personne et une file.
7. Spécifier sous forme de schémas les opérations partielles qui traitent les situations où une personne échoue à joindre une file, il s'agit des trois situations suivantes : la personne est déjà en file, la caisse vers laquelle veut aller la personne n'est pas en service ou la caisse ne fait pas partie du système.
8. Spécifier sous forme de schéma, l'opération *ChangerFilePartielle* qui décrit l'effet sur le système lorsqu'une personne décide de quitter une file et joindre une autre file d'attente. Cette opération prend en entrée, une personne et deux files.
9. Spécifier sous forme de schéma une opération partielle qui indique au système qu'une personne échoue de quitter une file et joindre une autre. La personne ne fait pas partie d'une file.
10. Spécifier les opérations totales *JoindreFileTotale* et *ChangementFileTotale*
11. Énoncer les théorèmes de l'initialisation et des pré-conditions