

# Specification Formelle

## Correction TD Schémas

Emna Menif

École Nationale de Carthage  
3<sup>eme</sup> Ingénierie des Systèmes Intelligents

A.U. 2021/2022

# Exercice1

$[EQUIPE, TEXTE]$

$ETAT\_COMP ::= inscription \mid tirage \mid tour \mid fin$

- Fonction *Tirage* :

$$\frac{}{Tirage : \mathbb{P} \mathbb{N} \rightarrow \mathbb{N}} \quad \forall e : \mathbb{P} \mathbb{N} \mid e \neq \emptyset \bullet (\exists n : \mathbb{N} \mid n \in e \bullet Tirage(e) = n)$$

- Fonction *Classe* :

$\langle A, B \rangle \text{ partition } C \Rightarrow A \cap B = \emptyset \wedge A \cup B = C$

$$\frac{}{Classe : \mathbb{P} \mathbb{N} \rightarrow \mathbb{P}(\mathbb{N} \rightarrow \mathbb{N})} \quad \forall e : \mathbb{P} \mathbb{N} \mid e \neq \emptyset \wedge \# e \bmod 2 = 0 \bullet \\ Classe(e) = \{f : \mathbb{N} \rightarrow \mathbb{N} \mid \langle \text{dom } f, \text{ran } f \rangle \text{ partition } e\}$$

# Suite exercice1

## Schéma d'état général *Competition*

### *Competition*

*inscrits* : seq *EQUIPE*

*etat* : *ETAT\_COMP*

*qualifies* :  $\mathbb{P}\mathbb{N}$

*matches* :  $\mathbb{N} \rightarrow \mathbb{N}$

*tours* : seq(( $\mathbb{N} \rightarrow \mathbb{N}$ )  $\rightarrow$  ( $\mathbb{N} \times$  *TEXTE*))

*qualifies*  $\subseteq$  dom *inscrits* (1)

dom *matches*  $\cap$  ran *matches* =  $\emptyset$  (2)

dom *matches*  $\subseteq$  dom *inscrits*  $\wedge$  ran *matches*  $\subseteq$  dom *inscrits* (3)

$\forall t : 1.. \# \text{tours} \bullet$  (dom(dom(*tours* *t*))  $\subseteq$  dom *inscrits*)  $\wedge$   
(ran(dom(*tours* *t*))  $\subseteq$  dom *inscrits*)  $\wedge$   
((dom(dom(*tours* *t*))  $\cap$  ran(dom(*tours* *t*))) =  $\emptyset$ )

## Suite exercice1

- *inscrits* : Équipes inscrites en tenant compte de l'ordre d'inscription
- *etat* : Donne l'état de la compétition
- *qualifies* : Équipes qualifiées pour le tour suivant
- *matches* : Matches à jouer le tour en cours (l'injection assure qu'une équipe joue une fois lors d'un tour)
- *tours* : Résultats des matchs joués lors des différent tours. Le codomaine est formé du numéro de l'équipe gagnante ainsi que du commentaire
- (1) Les qualifiés sont des équipes inscrites
- (2) Les équipes ne jouent qu'un seul match lors d'un tour
- (3) Les équipes qui jouent doivent être inscrites

## Suite exercice1

Schéma d'état initial *CompetitionInit*

*CompetitionInit*

*Competition*

*inscrits* =  $\emptyset$

*etat* = *inscription*

Les valeurs des autres variables se déduisent de *inscrits* selon le premier invariant

# Suite exercice1

## Schéma d'opération *Inscrire*

*Inscrire*

$\Delta$  *Competition*

$e? : EQUIPE$

$no! : \mathbb{N}$

$etat = inscription$  (1)

$e? \notin \text{ran } inscrits$  (2)

$inscrits' = inscrits \cap \langle e? \rangle$  (3)  $\wedge no! = \# inscrits'$  (4)

$etat' = etat \wedge qualifies' = qualifies \wedge matchs' = matchs \wedge tours' = tours$

- (1) la compétition n'a pas encore démarré
- (2) l'équipe n'est pas encore inscrite
- (3) l'équipe est inscrite et (4) le numéro séquentiel est retourné

# Suite exercice1

## Schéma d'opération *CloreInscription*

*CloreInscription*

$\Delta$  *Competition*

$etat = inscription$

$etat' = tirage$  (1)

$qualifies' = \text{dom } inscrits$  (2)

$inscrits' = inscrits \wedge matches' = matches \wedge tours' = tours$

- (1) Clore les inscriptions modifie l'état de la compétition pour permettre le tirage
- (2) Les équipes qualifiées pour le premier tour sont les équipes inscrites

# Suite exercice1

## Schéma d'opération *Tirage*

*Tirage*

$\Delta$  *Competition*

$etat = tirage$

$\# qualifies \bmod 2 = 0 \Rightarrow qualifies' = \emptyset$  (1)

$\# qualifies \bmod 2 = 1 \Rightarrow qualifies' = \{tirage(qualifies)\}$  (2)

$etat' = tour$  (3)

$tours' = tours \cap \langle \emptyset \rangle$  (3')

$(\exists f : \mathbb{N} \rightsquigarrow \mathbb{N} \mid f \in Classe(qualifies \setminus qualifies') \bullet matches' = f)$  (4)

$inscrits' = inscrits$

- (1) Si le nombre d'équipes est pair, il n'y aura pas d'équipe qualifiée d'office
- (2) Si le nombre d'équipes est impair, une équipe est qualifiée d'office
- (3) et (3') Un nouveau tour est alors engagé
- (4) Les matchs qui seront joués au tour suivant, auront lieu entre les équipes qualifiées du tour précédent sauf l'équipe qualifiée d'office



# Suite exercice1

## Schéma d'opération *Resultat*

### *Resultat*

$\Delta$  *Competition*

$e_1? : \mathbb{N}$

$e_2? : \mathbb{N}$

$g? : \mathbb{N}$

$r? : \text{TEXTE}$

$etat = tour$  (1)

$e_1? \mapsto e_2? \in matches$  (2)

$g? = e_1? \vee g? = e_2?$  (3)

$qualifies' = qualifies \cup \{g?\}$  (4)

$matches' = matches \setminus \{e_1? \mapsto e_2?\}$  (5)

$tours' = tours \oplus \{\# tours \mapsto (tours(\# tours) \cup$

$\{(e_1? \mapsto e_2?) \mapsto (g?, r?)\})\}$  (6)

$etat' = etat \wedge inscrits' = inscrits$

## Suite exercice1

- (1) L'opération n'est possible que lors du déroulement d'un tour
- (2) Le match joué doit être du tour courant
- (3) L'équipe gagnante est l'une des deux équipes qui jouent
- (4) L'équipe gagnante est qualifiée pour le tour suivant
- (5) Les deux équipes ne s'affronteront plus le tour en cours
- (6) Le résultat du match joué est enregistré dans les résultats du tournoi pour le tour courant

# Suite exercice1

Schéma d'opération *TourSuivant*

*TourSuivant*

$\Delta$  *Competition*

*etat* = *tour* (1)

*matches* =  $\emptyset$  (2)

$\# \text{qualifies} = 1 \Rightarrow \text{etat}' = \text{fin}$  (3)

$\# \text{qualifies} > 1 \Rightarrow \text{etat}' = \text{tirage}$  (4)

$\text{inscrits}' = \text{inscrits} \wedge \text{qualifies}' = \text{qualifies}$

$\text{matches}' = \text{matches} \wedge \text{tours}' = \text{tours}$

- (1) Un tour est en cours
- (2) Tous les matchs ont été disputés
- (3) Fin du tournoi s'il ne reste qu'un seul qualifié
- (4) Transition vers le tirage est déclenchée s'il y a plus d'un qualifié

# Suite exercice1

- Théorème d'initialisation :

**theorem** Théorème1

$\exists \textit{Competition} \bullet \textit{CompetitionInit}$

- Théorème de pré-condition de Inscire :

**theorem** Théorème2

$\forall \textit{Competition}, e? : \textit{EQUIPE} \mid \textit{etat} = \textit{inscription} \wedge e? \notin \textit{ran inscits}$   
• **pre** *Inscire*

**pre** *Inscire*

*Competition*  
*e? : EQUIPE*

$\exists \textit{Competition}', no! : \mathbb{N} \bullet (\textit{etat} = \textit{inscription} \wedge e? \notin \textit{ran inscits}$   
 $\textit{inscits}' = \textit{inscits} \cap \langle e? \rangle \wedge no! = \# \textit{inscits}'$   
 $\textit{etat}' = \textit{etat} \wedge \textit{qualifies}' = \textit{qualifies} \wedge$   
 $\textit{matches}' = \textit{matches} \wedge \textit{tours}' = \textit{tours})$

## Exercice2

[CAISSE, PERSONNE]

Schéma d'état général *Supermarche*

*Supermarche*

*Caisses* :  $\mathbb{P}$  *CAISSE* (1)

*Files* : *CAISSE*  $\rightarrow$  seq *PERSONNE* (2)

$\text{dom } Files \subseteq Caisses$

$\forall c_1, c_2 : \text{dom } Files \mid c_1 \neq c_2 \bullet \text{ran } Files(c_1) \cap \text{ran } Files(c_2) = \emptyset$  (3)

- (1) Les caisses du supermarché
- (2) Les files devant les caisses en service
- (3) Une personne ne peut pas être dans deux files différentes

## Suite exercice2

- Schéma d'état *PlusDeCaisse*

<i>PlusDeCaisse</i>	
<i>Supermarche</i>	
$\text{dom Files} = \text{Caisses}$	

- Schéma d'état *CaissesPleines*

<i>CaissesPleines</i>	
<i>Supermarche</i>	
$\forall c \in \text{dom Files} \bullet \#(\text{Files}(c)) \geq 3$	

- Schéma d'état initial *SupermarcheInit*

<i>SupermarcheInit</i>	
<i>Supermarche</i>	
$\text{dom Files} = \emptyset$	

## Suite exercice2

Schéma d'opération *JoindreFilePartielle*

*JoindreFilePartielle*

$\Delta$  *Supermarche*

$p? : \text{PERSONNE}$  (1)

$c? : \text{dom Files}$  (2)

$p? \notin \bigcup \{cs : \text{dom Files} \bullet \text{ran}(\text{Files}(cs))\}$  (3)

$\text{Files}' = \text{Files} \oplus \{c? \mapsto (\text{Files}(c?) \cap \langle p? \rangle)\}$  (4)

- (1) La personne qui veut joindre une file d'une caisse
- (2) La caisse concernée
- (3) La personne ne doit être dans aucune file
- (4) Les files des autres caisses restent inchangées et la personne rejoint la file de la caisse

## Suite exercice2

Schémas d'opération *DejaEnFile*, *CaisseFermee* et *CaisselInvalide*

*DejaEnFile1*

$\exists \text{Supermarche}$

$p? : \text{PERSONNE}$

$p? \in \bigcup \{c : \text{dom Files} \bullet \text{ran}(\text{Files}(c))\}$

*DejaEnFile2*

$\exists \text{Supermarche}$

$p? : \text{PERSONNE}$

$c? : \text{dom Files}$

$p? \in \text{ran}(\text{Files}(c?))$



## Suite exercice2

*CaisseFermee*

$\exists$  *Supermarche*

$c? : \text{Caisses}$

$c? \notin \text{dom } \textit{Files}$

*CaisseInvalide*

$\exists$  *Supermarche*

$c? : \text{CAISSE}$

$c? \notin \text{Caisses}$

## Suite exercice2

### Schémas d'opération *ChangerFilePartielle*

*ChangerFilePartielle*

$\Delta$ *Supermarche*

$p? : PERSONNE$  (1)

$c_1?, c_2? : \text{dom } Files$  (2)

$p? \neq \text{head}(Files(c_1?))$  (3)

$Files' = Files \oplus \{c_1? \mapsto Files(c_1?) \upharpoonright (\text{ran } Files(c_1?) \setminus \{p?\}),$

$c_2? \mapsto (Files(c_2?) \cap \langle p? \rangle)\}$  (4)

- (1) et (2)  $p?$  veut passer de la caisse  $c_1?$  à la caisse  $c_2?$
- (3)  $p?$  ne doit pas être en tête de la file de la caisse  $c_1?$
- (4) Les files des autres caisses restent inchangées,  $p?$  sort de la file de  $c_1?$  rejoint la file de  $c_2?$

## Suite exercice2

Schémas d'opération *NonEnFile*

*NonEnFile*

$\exists \text{Supermarche}$

$p? : \text{PERSONNE}$

$c? : \text{dom Files}$

$p? \notin \text{ran}(\text{Files}(c?))$

## Suite exercice2

Schémas d'opération *JoindreFileTotale*

$$\begin{aligned} \text{JoindreFileTotale} == & \text{JoindreFilePartielle} \vee \text{DejaEnFile1} \\ & \vee \text{CaisseFermee} \vee \text{CaisseInvalide} \end{aligned}$$

Schémas d'opération *ChangerFileTotale*

$$\begin{aligned} \text{ChangerFileTotale} == & \\ & \text{ChangerFilePartielle} \vee \text{DejaEnFile2}[c_2?/c?] \vee \text{NonEnFile}[c_1?/c?] \\ & \vee \text{CaisseFerme}[c_1?/c?] \vee \text{CaisseFerme}[c_2?/c?] \\ & \vee \text{CaisseInvalide}[c_1?/c?] \vee \text{CaisseInvalide}[c_2?/c?] \end{aligned}$$

## Suite exercice2

- Théorème d'initialisation :

**theorem** Théorème1

$\exists \text{ Supermarche} \bullet \text{ SupermarcheInit}$

- Théorème de pré-condition de JoindreFilePartielle :

**theorem** Théorème2

$\forall \text{ Supermarche}, p? : \text{PERSONNE}, c? : \text{dom Files} \mid$   
 $p? \notin \bigcup \{cs : \text{dom Files} \bullet \text{ran}(\text{Files}(cs))\}$   
• **pre** JoindreFilePartielle

**pre** JoindreFilePartielle

*Supermarche*

*p? : PERSONNE*

*c? : dom Files*

$\exists \text{ Supermarche}' \bullet p? \notin \bigcup \{cs : \text{dom Files} \bullet \text{ran}(\text{Files}(cs))\}$

$\text{Files}' = \text{Files} \oplus \{c? \mapsto (\text{Files}(c?) \cap \langle p? \rangle)\}$