

Exercice 1 (*mcfsl1-simpke*)

Soient deux ensembles A et B qui sont des parties de U .

- Ecrire un modèle Event-B qui utilise deux variables v et w deux sous-ensembles de A et B
- Ajouter une fonction partielle de A dans B .
- Définir un événement $e1$ qui transfère un élément de A dans B s'il n'est pas dans A .
- Définir un événement $e2$ qui crée un lien entre un élément de A et un élément de B .

Exercice 2 (*mcfsl1-variant*)

Un système permet de réaliser la somme de deux nombres $x0$ et $y0$ en ajoutant une unité à une variable z . Il comprend un événement $incx2z$ qui décroît la valeur de x d'une unité et qui augmente la valeur de z de une unité et un événement $incy2z$ qui décroît y d'une unité et qui augmente z d'une unité. Le processus global s'arrête quand les deux variables x et y sont nulles. Ecrire un modèle Event-B qui modélise ce système.

Exercice 3 (*mcfsl1-summation*)

Soit une suite de valeurs entières v_1, \dots, v_n où le nombre n est fixé. Ecrire une spécification événementielle décrivant le calcul de la somme des éléments de cette suite. Pour cela, vous devez décrire les données puis l'événement magique qui réalise ce calcul.

Exercice 4 (*mcsfi-ressources-pb1*),

Modéliser les problèmes suivants.

Question 4.1 (*mcsfi-ressources-pb1*)

On suppose disposer de ressources qui sont partagées par un ensemble de processus. Si un processus a besoin d'une ressource, il demande cette ressource et s'il n'a plus besoin de cette ressource, il la rend. Un processus peut utiliser plusieurs ressources à la fois mais une ressource ne peut pas être utilisée par deux processus à la fois.

Question 4.2 (*mcsfi-ressources-pb2*)

On suppose disposer de ressources qui sont partagées par un ensemble de processus. Si un processus a besoin d'une ressource, il demande cette ressource et s'il n'a plus besoin de cette ressource, il la rend. Un processus ne peut utiliser qu'une seule ressource à la fois et une ressource ne peut pas être utilisée par deux processus à la fois.

Exercice 5 (*mcfsl1-invariantssafety*)

Nous considérons le modèle suivant.

```

MACHINEM1
VARIABLES
   $x$ 
INVARIANTS
  ...
EVENTS
EVENT INITIALISATION
  BEGIN
     $act1 : x := -10$ 
  END
EVENT evt1
  WHEN
     $grd1 : x \geq -1$ 
  THEN
     $act1 : x := x+1$ 
  END
EVENT evt2
  WHEN
     $grd1 : x \leq -1$ 
     $grd2 : x \geq -44$ 
  THEN
     $act1 : x := x-1$ 
  END
END

```

On considère plusieurs cas pour l'invariant.

Question 5.1 (M1)

$inv1 : x \in \mathbb{Z}$
 $inv3 : x \leq -1$

Est-ce que toutes les conditions de vérification sont prouvées par le prouveur de l'application Rodin ? Expliquez clairement pourquoi elles sont prouvées ou non.

Question 5.2 (M2)

$inv1 : x \in \mathbb{Z}$
 $inv3 : x \leq -3$

Est-ce que toutes les conditions de vérification sont prouvées par le prouveur de l'application Rodin ? Expliquez clairement pourquoi elles sont prouvées ou non.

Question 5.3 (M3)

$inv1 : x \in \mathbb{Z}$
 $inv4 : -45 \leq x \wedge x \leq -10$

Est-ce que toutes les conditions de vérification sont prouvées par le prouveur de l'application Rodin ? Expliquez clairement pourquoi elles sont prouvées ou non.

Question 5.4 (M4)

$inv1 : x \in \mathbb{Z}$
 $inv3 : x \leq -3$
 $inv4 : -45 \leq x \wedge x \leq -10$
 $inv2 : x \leq -1$

Est-ce que toutes les conditions de vérification sont prouvées par le prouveur de l'application Rodin ? Expliquez clairement pourquoi elles sont prouvées ou non.

Exercice 6 (*alg-maxtwo numbers*)

Soit le contrat suivant annoté qui calcule le maximum de deux entiers naturels x_0 et y_0

Variables : X, Y, Z

Requires : $x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z}$

Ensures : $z_f = \max(x_0, y_0)$

$\ell_0 : \{x = x_0 \wedge y = y_0 \wedge z = z_0 \wedge x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z}\}$

if $X < Y$ **then**

$\ell_1 : \{x < y \wedge x = x_0 \wedge y = y_0 \wedge z = z_0 \wedge x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z}\}$

$Z := Y;$

$\ell_2 : \{x < y \wedge x = x_0 \wedge y = y_0 \wedge x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z} \wedge z = y_0\}$

else

$\ell_3 : \{x \geq y \wedge x = x_0 \wedge y = y_0 \wedge z = z_0 \wedge x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z}\}$

$Z := X;$

$\ell_4 : \{x \geq y \wedge x = x_0 \wedge y = y_0 \wedge x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z} \wedge z = x_0\}$

;

$\ell_5 : \{z = \max(x_0, y_0) \wedge x = x_0 \wedge y = y_0 \wedge x_0, y_0 \in \mathbb{N} \wedge z_0 \in \mathbb{Z}\}$

Algorithme 1: maximum de deux nombres non annotée

Question 6.1 Traduire l'automate de cet algorithme sous la forme d'une machine modifiant les variables x, y, z, pc .

Question 6.2 Valider la traduction en simulant quelques

Question 6.3 Ajouter les annotations et les pré et post conditions.

Question 6.4 Vérifier la correction partielle et l'absence d'erreurs à l'exécution.