



Une réponse correcte rapporte 2 points; sans réponse cela rapporte 0 point; au bout de deux réponses fausses, on retire 1 point à chaque réponse fausse nouvelle.

Nom:
Prénom:

Écrit

Question 1 On souhaite décrire un système gérant des ressources R partagées par un ensemble de processus P . On modélise la propriété suivante: Si une ressource r est utilisée par un processus p , alors elle ne peut pas être utilisée par un autre processus q distinct de p . Le modèle de ce système utilise une variable d'états $use \in R \mapsto P$.

```
CONTEXT CEX1
SETS P, R
END
```

```
MACHINE MEX1
SEES CEX1
VARIABLES
  use
INVARIANTS
  inv1 : use ∈ R ↦ P
EVENTS
INITIALISATION
  begin
    act1 : use := ∅
  end
requesting
  any
    r, p
  where
    grd1 : p ∈ P
    grd2 : r ∈ R
    grd3 : r ∉ dom(use)
  then
    act1 : use(r) := p
  end
releasing
  any
    r, p
  where
    grd1 : p ∈ P
    grd2 : r ∈ R
    grd3 : r ∈ dom(use)
  then
    use := use \ {r ↦ p}
  end
END
```

La question est la suivante: le modèle est correct et la propriété $inv1$ est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
X	

Question 2 On souhaite décrire un autre système gérant des ressources R partagées par un ensemble de processus P . On modélise la propriété suivante: Si une ressource r est utilisée par un processus p , alors elle ne peut pas être utilisée par un autre processus q distinct de p et un processus ne peut utiliser qu'une seule ressource à la fois. Le modèle de ce système utilise une variable d'états $use \in R \mapsto P$.

```

CONTEXT CEX1
SETS P, R
END

```

```

MACHINE MEX1
SEES CEX1
VARIABLES
  use
INVARIANTS
  inv1 : use ∈ R ↔ P
EVENTS
INITIALISATION
  begin
    act1 : use := ∅
  end
requesting
  any
    r, p
  where
    grd1 : p ∈ P
    grd2 : r ∈ R
    grd3 : r ∉ dom(use)
  then
    act1 : use(r) := p
  end
releasing
  any
    r, p
  where
    grd1 : p ∈ P
    grd2 : r ∈ R
    grd3 : r ∈ dom(use)
  then
    use := use \ {r ↦ p}
  end
END

```

La question est la suivante: le modèle est correct et la propriété *inv1* est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
	x

Question 3 La propriété de correction partielle est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
x	

Question 4 On souhaite décrire un autre système gérant des ressources R partagées par un ensemble de processus P . On modélise la propriété suivante: Si une ressource r est utilisée par un processus p , alors elle ne peut pas être utilisée par un autre processus q distinct de p et un processus ne peut utiliser qu'une seule ressource à la

fois. Le modèle de ce système utilise une variable d'états $use \in R \mapsto P$.

```

CONTEXT CEX1
SETS P, R
END

```

```

MACHINE MEX1
SEES CEX1
VARIABLES
  use
INVARIANTS
  inv1 : use ∈ R ↔ P
EVENTS
INITIALISATION
  begin
    act1 : use := ∅
  end
requesting
  any
    r, p
  where
    grd1 : p ∈ P
    grd2 : r ∈ R
    grd3 : r ∉ dom(use)
    grd3 : p ∉ ran(use)
  then
    act1 : use(r) := p
  end
releasing
  any
    r, p
  where
    grd1 : p ∈ P
    grd2 : r ∈ R
    grd3 : r ∈ dom(use)
  then
    use := use \ {r ↦ p}
  end
END

```

La question est la suivante: le modèle est correct et la propriété *inv1* est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
	x

Question 5 La propriété de correction partielle est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
x	

Question 6 On suppose qu'un algorithme ou un programme est annotée au point de contrôle ℓ par la propriété $P_\ell(v)$ où v est la variable du programme. On suppose que les conditions de vérification ont été vérifiées. La propriété $pc = \ell \Rightarrow P_\ell(v)$ est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
x	

Question 7 La propriété énonçant qu'au plus 19 personnes peuvent être admises dans une salle avec une seule porte est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
x	

Question 8 La propriété suivante est une propriété de sûreté: Le niveau de l'eau de la citerne est au moins de 1 mètre.

Vous mettez une croix dans les cases appropriées.

vrai	faux
x	

Question 9 La propriété suivante est une propriété de sûreté: Le niveau de l'eau de la citerne atteindra le niveau de 5 mètres.

Vous mettez une croix dans les cases appropriées.

vrai	faux
	x

Nous considérons le modèle suivant.

```

MACHINEM1
variables
  x
invariants
  inv1 :  $x \in \mathbb{Z}$ 
  inv2 :  $x \leq 0$ 
  inv3 : ???
EVENTS
INITIALISATION
  begin
    act1 :  $x := -8$   end
  evt1
  when
    grd1 :  $x \geq 0$ 
  then
    act1 :  $x := x + 1$ 
  end
  evt2
  when
    grd1 :  $x \leq 0$ 
    grd2 :  $x \geq -12$ 
  then
    act1 :  $x := x - 1$ 
  end
end
END

```

Question 10 Toutes les conditions de vérification sont prouvées par le prouveur de l'application Rodin.

Vous mettez une croix dans les cases appropriées.

vrai	faux
	x

Question 11 inv2 est une propriété de sûreté.

Vous mettez une croix dans les cases appropriées.

vrai	faux
x	

Question 12 inv2 est une propriété invariante..

Vous mettez une croix dans les cases appropriées.

vrai	faux
	x

Question 13 inv3 est une propriété à ajouter pour que le modèle soit correct mais peut-être TRUE selon les réponses aux questions précédentes.

Vous écrivez inv3 dans la case ci-dessous:

$-13 \leq x \wedge x \leq -8$

Nous considérons le modèle suivant.

```

CONTEXT EC
SETS
  S, L
CONSTANTS
  l1, l2, A, B, x
AXIOMS
  axm1 : partition(L, {l1}, {l2})
  axm2 : A ∈ S
  axm3 : B ∈ S
  axm4 : x ∈ S
  axm5 : x = A
  axm7 : S ⊆ {A, B}
END

```

```

MACHINEM1
SEESEC
variables
  y, pc
invariants
  inv1 : pc ∈ L ∧ y ∈ S
  inv2 : pc = l1 ⇒ y ∈ S (Floyd's annotation)
  inv3 : pc = l2 ⇒ y = B (Floyd's annotation)
  inv4 : x ∈ S ∧ x = A ∧ y ∈ S ⇒ y ∈ S (checking precondition)
  inv5 : y = B ⇒ y ∈ S ∧ y = B (checking postcondition)
EVENTS
INITIALISATION
  begin
    act1 : y := S
    act2 : pc := l1
  end
e
  when
    grd1 : pc = l1
  then
    act1 : pc := l2
    act2 : y : |(y' ∈ S ∧ y' ≠ x)
  end

```

Question 14 *Ecrire la relation $BA(e)(y, pc, y', pc')$
Vous écrivez la relation dans la case ci-dessous:*

$BA(e)(y, pc, y', pc') \stackrel{def}{=} \\
pc = l1 \\
pc' = l2 \\
(y' \in S \wedge y' \neq x)$

Question 15 *Ecrire la condition exprimant que l'événement e préserve l'invariant de cette machine, noté $I(y, pc)$.*

Vous écrivez la condition dans la case ci-dessous:

$I(y, pc) \wedge BA(e)(y, pc, y', pc') \Rightarrow I(y', pc')$

Question 16 *Ecrire la condition exprimant que l'événement e est faisable.*
Vous écrivez la condition dans la case ci-dessous:

$$I(y, pc) \wedge pc = l1 \Rightarrow \exists y', pc'. BA(e)(y, pc, y', pc')$$

Question 17 La machine $M1$ n'est pas prouvée complètement. Il manque une preuve que l'événement est faisable. Pour que cet événement soit faisable, il faut ajouter un axiome et vous devez quel est l'axiome à ajouter.

Vous mettez une croix dans les case appropriées.

$axm6 : A = B$	$axm6 : A \neq B$
	x

On décide de modéliser la gestion de services S par des utilisateurs P

Question 18 Ecrire une fonction souscription qui enregistre les services souscrits par chaque personne de P . Vous écrivez la fonction dans la case ci-dessous:

$inv1 : fsous \in SERVICES \rightarrow POWER(PERSONNES)$

Question 19 Ecrire un événement de souscription à un service nouveau.

Vous écrivez l'événement dans la case ci-dessous:

```
souscription
any
s, p  where
  grd1 : s ∈ SERVICES
  grd2 : p ∈ PERSONNES
  grd3 : p ∉ fsous(s)
then
  act2 : fsous(s) := fsous(s) ∪ {p}
end
```

Fin de l'énoncé