

Théorie et pratique de la concurrence

Contrôle Continu

**Exercice 1 [2 points]** Soit une fonction  $f$  pour laquelle il existe une valeur  $i$  telle que  $f(i) = 0$ . Les algorithmes concurrents ci-dessous cherchent une telle valeur  $i$  (appelée aussi le

point zéro) en divisant l'espace de recherche en deux parties : les points zéro positifs  $i \geq 0$  et les points zéro négatifs  $i < 0$ . Un algorithme est correct si, pour toutes les exécutions, les

- (c) "2 apparaît une infinité de fois" *10 points*  
 (d) "2 ne peut apparaître qu'après 0" *10 points*  
 4. Pour chaque propriété ci-dessus, indiquez si le programme concurrent la vérifie ou non.

**Exercice 3 [5 points]** Soit la solution suivante pour l'implémentation d'une section critique :

### Algorithme 1

```
boolean wantp = False, wantq = False; // variables partagées
```

```
Process P
loop forever:
p1: section NC
p2: await (wantq==false)
p3: wantp:=True
p4: section critique
p5: wantp:=False
```

```
Process P
loop forever:
p1: section NC
p2: await (wantp==false)
p3: wantq:=True
p4: section critique
p5: wantq:=False
```

1. Construisez le diagramme d'états de cet algorithme
2. Quelles sont les propriétés vérifiées par cet algorithme (exclusion mutuelle, absence de famine, absence d'interblocage) ?

**Exercice 4 [5 points]** Considérons l'Algorithme 2 pour l'implémentation d'une section critique :

### Algorithme 2

```
int wantP = 0, wantQ = 0; // variable partagée
```

```
Process P
loop forever:
  1 if (wantQ = 1)
  2   wantP := -1
  3 else
  4   wantP := -1
  5   await (wantP != wantQ)
  6   section critique
  7   wantP := 0
```

```
Process Q
loop forever:
  1 if (wantP = -1)
  2   wantQ := 1
  3 else
  4   wantQ := -1
  5   await (wantQ == wantP)
  6   section critique
  7   wantQ := 0
```

1. Numérotez les lignes de cet algorithme dans l'idée de pouvoir construire ensuite le diagramme d'états
2. Exprimez à l'aide d'une formule LTL la propriété d'exclusion mutuelle
3. Exprimez à l'aide d'une formule LTL la propriété d'absence de famine
4. En utilisant le diagramme d'états, indiquez si cet algorithme vérifie la propriété d'exclusion mutuelle (dans le cas d'une réponse négative, seulement une partie du diagramme d'états exhibant le comportement incorrect est demandée).

La P... 1874... la liste des