INSTN EXAMEN-SETI 03/12/2019

Sans document et sans calculatrice

V. David, S. Louise, F. Thomas

Cet examen (2h45) est constitué de trois parties indépendantes.

PARTIE A

Systèmes asynchrones

Question de cours (3 points)

- **Q1 :** Définir le rôle et le fonctionnement d'une instruction de type « TAS » (Test And Set), et pour quelle type d'architecture de processeur elle est utile.
- **Q2 :** Dans le cas d'une architecture multiprocesseur à bus partagé, sans instruction de type « TAS », décrire de manière détaillée une solution logicielle équivalente.
- **Q3 :** Définir l'atomicité logicielle et la sériabilité.
- **Q4 :** Définir la famine et l'interblocage. Donnez des exemples et présentez au moins une méthode pour garantir l'absence d'interblocage.

Sans document et sans calculatrice

Problème : synchronisation avec les sémaphores (3 points)

Problème 1 :

Trois tâches T1, T2 et T3 pilotent chacune un appareil qui utilise la même aire de travail.

Ecrire une synchronisation entre T1, T2 et T3 qui garantisse que l'aire de travail n'est utilisée que par un seul appareil à la fois.

Problème 2 :

Trois tâches P, C1 et C2 pilotent trois appareils. L'appareil piloté par P dépose des pièces sur un plateau central fixe à 5 places. Les appareils pilotés respectivement par C1 et C2 prennent une pièce sur le plateau et la dépose sur un poste d'usinage avec une capacité de stockage limité à 8.

Ecrire les données utilisées et les synchronisations nécessaires entre P, C1 et C2 qui garantissent que quand P lance son appareil, il est certain qu'il y a une place disponible identifiée sur le plateau et que quand C1 ou C2 lance leur appareil, il est certain qu'il y a au moins une pièce sur une place identifiée du plateau central. P, C1 et C2 doivent travailler en parallèle chaque fois que cela est possible.

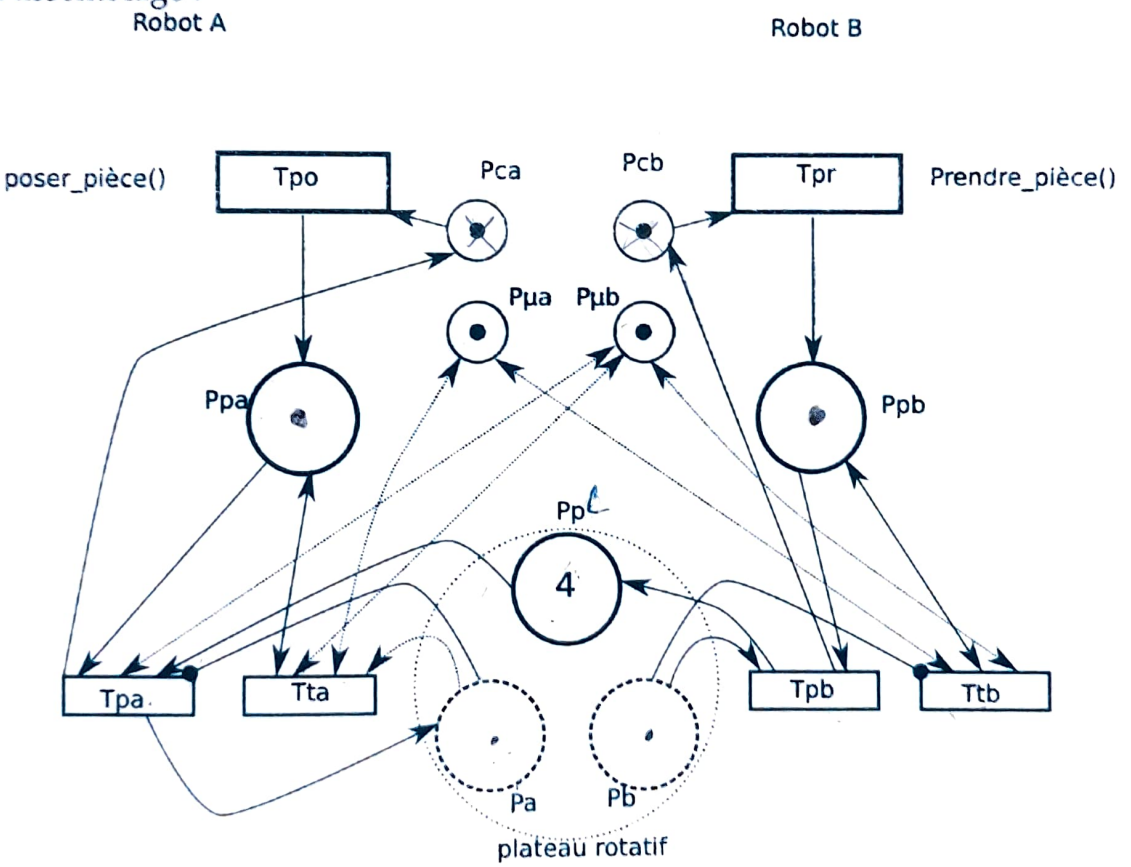
Enoncez et expliquez les propriétés démontrant le bon fonctionnement des trois appareils ensembles, en termes de performance et de sûreté.

Sans document et sans calculatrice

PARTIE B

Modélisation et analyses fondées sur les réseaux de Petri (5 points)

On considère le réseau de Petri suivant correspondant au problème simplifié de la chaîne de montage avec deux robots A et B, le robot A posant des pièces sur un poste d'assemblage à plateau rotatif et le robot B reprenant les pièces du poste d'assemblage :



Sans document et sans calculatrice

On suppose que les transitions Tpo et Tpr sont des transitions commandées par les appels aux fonctions *poser_piece*() et *prendre_piece*() respectivement pour les robots A et B.

Graphe de marquage

On suppose que le système est en plein fonctionnement. L'appel aux deux fonctions mentionnées précédemment ont été réalisées, et de ce fait les marquages sont modifiés de la façon suivante :

- le marquage en Ppa et en Ppb vaut 1
- le marquage en Pca et en Pcb vaut 0
- le marquage en Ppl vaut 3 et il vaut 1 en Pa

En prenant ces valeurs pour le marquage initial (les autres étant inchangées), développer le graphe de marquage du réseau de Petri. On fera l'hypothèse supplémentaire pour cela que la rotation du plateau (Tta ou Ttb) inversera les jetons entre Pa et Pb s'il y en a.

Algèbre linéaire

Donnez la matrice d'incidence. Pour simplifier, on ne prendra pas en compte les places Pa et Pb ainsi que les arcs qui les connectent.

Montrez qu'il existe au moins deux invariants de marquage. Il peut en exister un troisième en prenant en compte correctement Pa et Pb mais cela n'est pas utile de le démontrer.

Sans document et sans calculatrice

Programmation

A l'aide des appels systèmes *debut_atomique*() et *fin_atomique*() et les fonctions supplémentaires *tourner_plateau*() et *deposer_piece_sur_plateau*(), écrire la fonction *poser_piece*() qui modélise le comportement du robot A.

Pourquoi les places Pμa et Pμb sont-elle nécessaires ?

PARTIE C

Sûreté de fonctionnement des systèmes critiques

Questions à Choix Multiples (5 points)
(Entourez la ou les bonnes réponses pour chacune des questions)

Sur quels critères pouvez-vous conclure que le système à concevoir est compliqué et par conséquent qu'il nécessite des approches de conceptions particulières ?

1. Environnement dynamique (lié au temps)
2. Système séquentiel
3. Système modulaire
4. Composants hétérogènes
5. Exécutions concurrentes et simultanéités

Quand peut-on avoir confiance dans le développement d'un système ?

1. Quand il est déterminisme
2. Quand il y a du partitionnement temporel
3. Quand il y a du partitionnement spatial
4. Quand il est certifié/qualifié
5. Quand tous ces tests sont OK et nombreux
6. Quand il met en œuvre des APIs POSIX ou ARINC-653

Sans document et sans calculatrice

Le partitionnement spatial c'est le regroupement des composants dans un même espace mémoire pour assurer de manière sûr qu'un composant impact un autre composant.

1. Vrai
2. Faux

Le partitionnement temporel c'est la réservation du processeur par priorité. L'objectif est d'assurer qu'une fonction aura tout le temps nécessaire pour s'exécuter.

1. Vrai
2. Faux

La certification DO-178C impose les méthodes de développement et les objectifs à atteindre pour le développement d'un système sûr de fonctionnement.

1. Vrai
2. Faux

Dans la règlementation avionique DO-178C, la vérification est l'activité pour répondre à la question : "Est-ce que le logiciel est bien réalisé ?" ?

1. Vrai
2. Faux

L'exigence suivante est testable :
"Mon Disjoncteur doit couper le courant sur la sortie Relai si la valeur mesurée sur l'entrée Capteur dépasse strictement 16 Ampères."

1. Vrai
2. Faux

Dans une approche Event-Trigger, le sémaphore est un exemple d'événement déclenchant une exécution.

1. Vrai
2. Faux

Dans une approche Event-Trigger, nous ne devons pas faire d'hypothèses sur l'ordonnancement pour concevoir notre système.

1. Vrai
2. Faux

Sans document et sans calculatrice

Dans une approche de conception Time-Triggered les données sont visibles dès qu'elles sont produites.
1. Vrai
2. Faux

Dans une approche Time-Triggered, des deadlocks peuvent survenir à l'exécution.
1. Vrai
2. Faux

AADL est un langage pour décrire l'architecture statique multitâche d'un programme (tâches, propriétés temporelles)
1. Vrai
2. Faux

Le langage PsyC est un langage déclaratif qui permet de concevoir des logiciels multitâches partitionnés dirigés par les événements.
1. Vrai
2. Faux

Entourez les réponses vraies
1. Le noyau d'un système d'exploitation s'exécute en mode utilisateur
2. Pour passer d'un mode user à un mode privilégié, des appels systèmes sont mis en place dans les APIs du système d'exploitation
3. Posix propose de quoi faire du partitionnement temporel
4. Un process POSIX possède par défaut un thread
5. Un process ARINC est une partition spatiale

Questions de cours (4 points)

1. Expliquez les étapes et les documents produits pour développer un système temps réel sûr de fonctionnement dans le contexte de la norme DO-178C.

2. En l'illustrant par un schéma, expliquez la différence entre les approches dirigées par le temps et les approches dirigées par les événements

Sans document et sans calculatrice

(interruptions) ? Quel est l'avantage d'une conception événementiel par interruption par rapport à une conception dirigée par le temps ?

3. Citez un mécanisme matériel pour réaliser la protection mémoire et un mécanisme logiciel présent dans la norme Posix pour faire du partitionnement mémoire.

4. La fonction ci-dessous est-elle réentrante ? Est-elle threadsafe ? Pourquoi ?

```
void swap(int *x, int *y) {
    static int t;
    t = *x;
    printf(« %i\n »,t);
    *x = *y;
    *y = t;
}
```