

Examen semestriel

Algorithmique et Systèmes d'exploitation distribués (Corrigé)

1 H30

**Exercice 1 (06 points)** : La détection de la terminaison d'un calcul est l'un des problèmes étudiés dans les systèmes répartis.

**Question 1** : Expliquez en quelques lignes pourquoi ce problème est réputé difficile.

Réponse : Dans un système réparti, un calcul peut être fait par plusieurs processus situés sur des sites différents. Pour savoir s'il y'a terminaison du calcul , il faut examiner l'état de tous les processus concernés et les éventuels messages en transit à travers le réseau.

(1pt)

**Question 2** : Pour modéliser ce problème, on considère que chaque processus est soit dans l'état actif, soit dans l'état passif, à un instant donné. Expliquez ce que signifie chaque état.

Réponse :

Etat actif : le processus est en exécution.

Etat passif : le processus est terminé , ou il est en attente d'un message.

(0.5pt)

**Question 3** : Quand, un processus peut-il passer de l'état actif à l'état passif ?.

Réponse :

Un processus passe de l'état actif à l'état passif : s'il a terminé son exécution ou s'il se met en attente d'un message.

(0.5pt)

**Question 4** : Quand, un processus peut-il passer de l'état passif à l'état actif ?.

Réponse :

Un processus passe de l'état passif à l'état actif : s'il reçoit un message.

(0.5pt)

**Question 5** : A un instant donné, on constate que tous les processus du système sont dans un état passif. Peut-on dire qu'il y'a terminaison ? Justifiez.

Réponse :

Non. Si tous les processus sont dans un état passif, cela n'implique pas qu'il y'a terminaison de processus , car il se peut qu'il y'ait un message en transit dans le réseau qui n'a pas été reçu par le destinataire. En le recevant, le destinataire redeviendra actif, et il va falloir tout recommencer. Pour qu'il y'ait terminaison il faut que deux conditions soient réunies :

Tous les processus doivent être dans un état passif + Il n'ya aucun message en transit dans le réseau.

(0.5pt)

Pour résoudre ce problème, un algorithme (vu en cours) organise les processus en un anneau virtuel unidirectionnel, sur lequel circule un jeton. On suppose que chaque processus est sur un site différent.

**Question 6** : Que contient ce jeton ?.

Réponse :

Le jeton contient le nombre de processus passifs comptés pendant sa circulation dans l'anneau.

(0.5pt)

**Question 7** : Le jeton est coloré en blanc ou en noir . Que signifie chaque couleur ? .

Réponse :

Blanc : Tous les processus visités sont passifs, et leur état n'a pas changé depuis le dernier passage du jeton.

Noir : Il y'a au moins un processus qui est passé de Passif à Actif , depuis le dernier passage du jeton.

(1pt)

**Question 8** : Quelle est le traitement à faire lorsque le jeton arrive sur un site ? .

Réponse :

Lorsque le processus (présent sur le site) reçoit le jeton, il attend d'être passif avant d'envoyer le jeton. Il transmet le jeton en fonction du traitement qu'il vient de faire (a-t-il envoyé un message ou pas).

Si le processus n'a pas envoyé un message à un autre site

Alors Il incrémente la valeur du jeton ,

Il transmet le jeton sans modifier la couleur (blanche) de ce dernier

Sinon //Le processus a envoyé un message

Il réinitialise la valeur du jeton

Il transmettra le jeton en noir.

Finsi

(1pt)

**Question 9** : Avec cet algorithme, quand pouvons-nous dire que nous avons une terminaison de calcul ?.

Réponse :

Il y'a terminaison si le jeton revient blanc avec le compteur égale à N (Tous les sites ont été visités)

(0.5pt)

---

**Exercice 2 (06 points)** On considère un système réparti composé de 5 sites, ayant chacun une priorité différente :  $S5 > S4 > S3 > S2 > S1$ . Le site 5 est le coordinateur.

**Question 1** : Que doit-on faire si on détecte la panne du site 2 ? .

Réponse :

On doit informer les autres sites pour qu'ils actualisent leurs tables de routage.

Mais, il n'y a pas une élection de coordinateur.

(0.5pt)

**Question 2** : Que doit-on faire si on détecte la panne du site 5 ? .

Réponse :

On doit lancer le processus d'élection d'un nouveau coordinateur.

(0.5pt)

On utilise l'algorithme "Brutal" vu en cours pour l'élection d'un nouveau coordinateur.

**Question 3** : Si c'est le site 1 qui a détecté la panne du coordinateur, combien de messages sont nécessaires pour désigner un nouveau coordinateur. Justifiez.

Réponse :

- S1 envoie des messages d'élection à S2, S3 et S4.

- S2 envoie des messages d'élection à S3 et S4.

- S3 envoie un message d'élection à S4.

- Après sa désignation comme nouveau coordinateur, S4 envoie des messages de proclamation à S1, S2 et S3.

Au total, nous avons : 6 messages d'élection + 3 messages de proclamation = 9 messages.

(1pt)

**Question 4** : Même question que précédemment, mais en supposant que c'est S3 qui a détecté la panne du coordinateur.

Réponse :

- S3 envoie un message d'élection à S4.

- Après sa désignation comme nouveau coordinateur, S4 envoie des messages de proclamation à S1, S2 et S3.

Au total, nous avons : 1 message d'élection + 3 messages de proclamation = 4 messages.

(1pt)

**Question 5** : A partir des questions 3 et 4 précédentes, trouvez un résultat général ?

Réponse :

La priorité du site qui a détecté la panne du coordinateur influe sur le nombre de messages : plus cette priorité est proche de celle du coordinateur, moins seront les messages d'élection. Mais, le nombre de messages de proclamation est le même dans tous les cas.

Soit Si le site du a détecté la panne du coordinateur , et n la priorité du coordinateur :

Le nombre de messages d'élection =  $(n - i) + (n - i - 1) + (n - i - 2) + \dots + 1$

Le nombre de messages de proclamation =  $n - 1$

(1.5 pt)

**Question 6** : Que fait-on si le site S2 (tombé en panne) reprend son service ? . Justifiez.

Réponse :

Le site relance l'élection .

(0.5pt)

**Question 7** : Que fait-on si le site S5 (tombé en panne) reprend son service ? . Justifiez.

Réponse :

Le site relance le processus d'élection .

(0.5pt)

**Question 8** : Pourquoi cet algorithme est appelé "Brutal" ?

Réponse :

Il est appelé brutal (bully en anglais) car si l'ancien site coordinateur reprend son service, il relance le processus d'élection : il rétrograde le coordinateur actuel et se proclame nouveau coordinateur.

(0.5pt)

---

**Exercice 3 (08 points)** : Un système réparti est composé de 3 sites S1, S2 et S3. Les sites S2 et S3 veulent entrer en section critique lorsque leurs horloges logiques sont égales respectivement à 4 et 2. On utilise l'algorithme RicartAgrawala.

**Question 1** : Que fait le site 2, lorsqu'il reçoit la requête du site 3 ? .

Réponse :

Il répond avec un Ack, car la requête du site 3 a une valeur d'estampille ( $h_3=3$ ) inférieure à la date de la requête du site 2 lui-même ( $h_2=4$ ).

(1pt)

**Question 2** : Que fait le site 3, lorsqu'il reçoit la requête du site 2 ? .

Réponse :

Il ne répond pas. Il met la requête du site 2 dans sa file, car la requête du site 2 a une valeur d'estampille ( $h_2=4$ ) supérieure à la date de la requête du site 3 lui-même ( $h_3=2$ ).

(1pt)

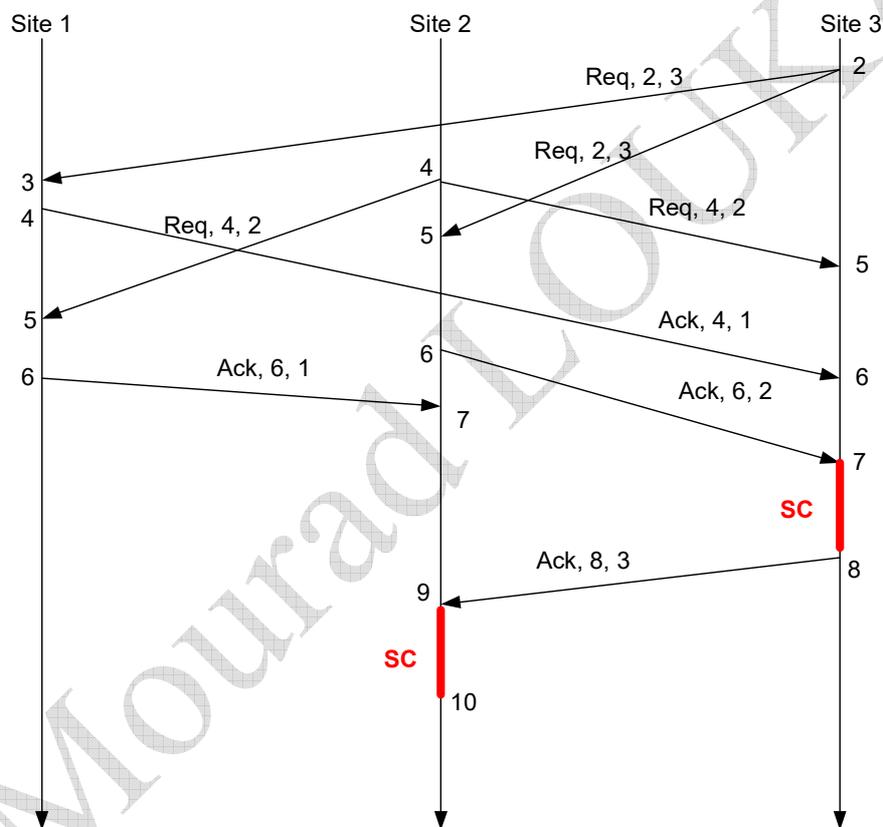
**Question 3** : Que contient la file du site 1, au fil du temps?.

Réponse :

La file du Site 1 est vide à tout temps, car ce site n'a pas fait de requête pour entrer en SC. Il se contente de répondre par un ACK à chaque requête reçue.

(1pt)

**Question 4** : Représentez sur un schéma le déroulement complet de l'algorithme.



(2pt)

On suppose maintenant que nous avons sur les sites S1, S2 et S3 respectivement les processus P1, P2 et P3 suivants. SC désigne la demande d'entrée en Section critique. k désigne un entier supérieur à 1.

```

P1 (site 1)
Début
    Pour i:= 1 jusqu'à k
    Faire
        .....
        SC
        .....
    Fait
Fin.

```

```

P2 (site 2)
Début
    .....
    SC
    .....
Fin.

```

```

P3 (site 3)
Début
    Si (condition) Alors
        .....
    Sinon
        .....
        SC
        .....
    Finsi
Fin.

```

**Question 5** : Combien de messages au total sont nécessaires pour dérouler les 3 processus, avec l'algorithme Ricart-Agrawala ? . Justifier.

Le déroulement d'une seule SC exige 2 messages Req et 2 messages Ack.

L'exécution ou non de la SC du processus P3 dépend de la valeur de la condition qui existe dans son code.

Le tableau suivant résume le nombre de messages nécessaires pour chaque processus.

	Nombre de message Req	Nombre de message Ack
Processus P1	$K * 2$	$K * 2$
Processus P2	2	2
Processus P3		
Si la condition existant dans le code du processus est vraie	0	0
Si elle est fausse	2	2

(3pt)