

UNIVERSITÉ DE SHERBROOKE  
DÉPARTEMENT D'INFORMATIQUE

IFT 630

Processus concurrents et parallélisme

Examen final

Le jeudi 15 avril 2021  
de 9 h à 12 h

Professeur : Gabriel Girard

- Notes :
- Toute documentation est permise.
  - Répondez dans les espaces prévus à cet effet.
  - Cet examen comporte 10 questions sur 20 pages.
  - Le total de l'examen est sur 110. Le résultat sera tronqué sur 100.
  - **Justifiez chacune de vos réponses.**

Nom : \_\_\_\_\_ Prénom : \_\_\_\_\_

Signature : \_\_\_\_\_ CIP : \_\_\_\_\_

Question	Barème
1	/ 9
2	/ 5
3	/ 12
4	/ 6
5	/ 10
6	/ 15
7	/ 18
8	/ 15
9	/ 10
(Bonus) 10	/10
Total :	/110

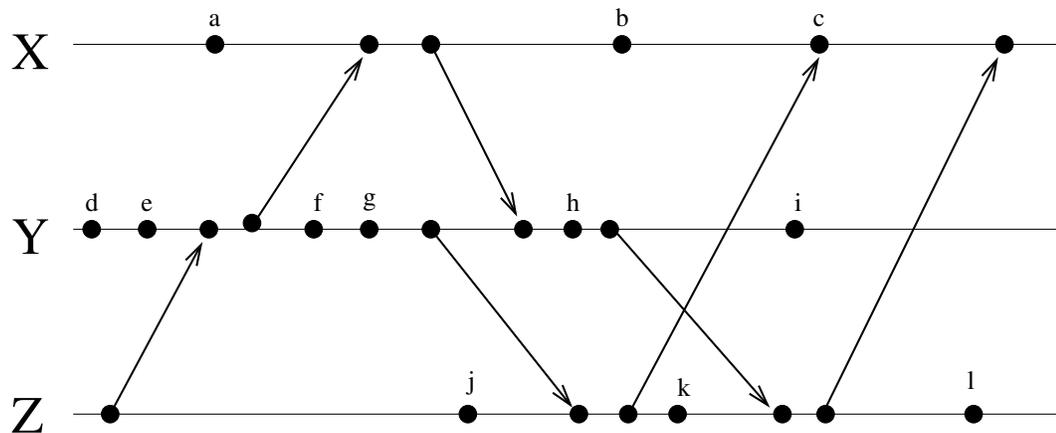




### 3. Les horloges (12 points)

(a) Les horloges logiques de Lamport

La figure suivante montre un certain nombre d'événements se produisant entre trois sites X, Y et Z. Que peut-on dire sur les événements  $a, c, f, i, j$  et  $l$  selon la relation de précédence? En supposant que les horloges sont initialisées à 0 et que les événements sont seulement des envois et des réceptions de messages ainsi que les événements locaux indiqués sur la figure, quelles sont les valeurs d'horloge associées aux événements  $a, c, f, i, j$  et  $l$ ? Servez-vous de la figure pour illustrer les différentes valeurs associées à l'horloge.




---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---





5. **Vrai ou Faux ? Justifiez chacune de vos réponses. (10 points)**

(a) Les liens de communication sont logiquement équivalents aux sémaphores.

---

---

---

---

---

---

---

(b) Il n'y a aucune différence entre une variable condition et une sémaphore.

---

---

---

---

---

---

---

(c) Lorsqu'une erreur est détectée, on sait automatiquement quelle est la faute qui l'a causée.

---

---

---

---

---

---

---

(d) Un système correct est automatiquement fiable.

---

---

---

---

---

---

---

(e) Il est possible de concevoir une méthode généralisée de reprise avant.

---

---

---

---

---

---

---





## 7. Le carrefour (18 points)

La circulation au carrefour de deux voies est réglée par des feux verts et rouges. Quand le feu est vert pour une voie, les voitures y circulant peuvent traverser le carrefour ; quand le feu est rouge, elles doivent attendre (on admet que les voitures de chaque voie traversent le carrefour en ligne droite seulement). On impose les conditions suivantes :

- i. toute voiture se présentant au carrefour doit le franchir au bout d'un temps fini ;
- ii. les feux de chaque voie passent alternativement du vert au rouge, chaque couleur étant maintenue pendant un temps fini ;
- iii. un instant donné, le carrefour ne doit contenir que des voitures d'une même voie.

Les arrivées sur les deux voies sont distribuées de façon quelconque. Le fonctionnement de ce système est représenté par un ensemble de processus parallèles. Un processus «CHANGEMENT» gère la commande des feux et un processus particulier est associé à chaque voiture. La traversée du carrefour par une voiture de la voie  $i$  ( $i = 1,2$ ) correspond à l'exécution d'une procédure «TRAVERSEE  $i$ » par le processus associé à cette voiture. On demande d'écrire le programme du processus «CHANGEMENT» et des procédures «TRAVERSEE 1» et «TRAVERSEE 2» dans les deux cas suivants :

- (a) le carrefour peut contenir une seule voiture à la fois ;
- (b) le carrefour peut contenir au plus  $K$  voitures à la fois.

Le fonctionnement correct des feux doit être maintenu quel que soit l'ordre d'arrivée des voitures. La modification des feux par changement doit donc comporter les opérations suivantes :

- interdire aux nouvelles voitures arrivant sur la voie où le feu est vert de s'engager dans le carrefour (pour respecter la condition ii) ;
- attendre que les voitures engagées dans le carrefour en soient sorties avant d'ouvrir le passage sur l'autre voie (pour respecter la condition iii).

Dans le cas (a), on a un problème d'exclusion mutuelle. Dans le cas (b), on pourra remarquer que le problème est analogue à celui des lecteurs et des écrivains, et adopter une solution du même type. Pour assurer la synchronisation, utilisez les moniteurs.

---

---

---

---

---





















## Tri Sélection (extrait de Wikipédia)

Sur un tableau de  $n$  éléments (numérotés de 0 à  $n-1$ ), le principe du tri par sélection est le suivant :

1. rechercher le plus petit élément du tableau, et l'échanger avec l'élément d'indice 0 ;
2. rechercher le second plus petit élément du tableau, et l'échanger avec l'élément d'indice 1 ;
3. continuer de cette façon jusqu'à ce que le tableau soit entièrement trié.

En pseudo-code, l'algorithme s'écrit ainsi :

```
1 tri_selection(tableau t)
2   n := longueur(t)
3   pour i de 0 à n - 2
4     min := i
5     pour j de i + 1 à n - 1
6       si t[j] < t[min], alors min := j
7     fin pour
8     si min != i, alors échanger t[i] et t[min]
9   fin pour
10 fin tri_selection
```