

Révision finale



GIF-1001 Ordinateurs: Structure et Applications, Hiver 2018
Jean-François Lalonde

Logistique: examen final

- 40% de la note totale
- Mardi 1 mai de 14h30 à 17h30
- Local: PLT-1112
- Une feuille 8.5 x 11, recto-verso, écrite *à la main*
- Droit à la calculatrice (approuvée)
- Examen récapitulatif
 - Pas de question de programmation ARM
 - Cependant, vous devez pouvoir *comprendre* et du code ARM

Stratégies *avant* l'examen

- Dans les jours précédant l'examen:
 - DORMIR
 - Étude:
 - exercices facultatifs, exercices faits en classe
 - examens des années précédentes
 - lecture: notes de cours, livre
 - objectifs du cours (sur le site web)
- Dans les heures précédant l'examen:
 - Alimentation adéquate (attention au «sugar crash»)
 - Économisez votre énergie intellectuelle
 - Détente

Stratégies *durant* l'examen

- *Avant* de commencer:
 - Compter les pages
 - Arracher la feuille des annexes pour un accès facile
 - Survoler l'examen et classer les questions selon leur difficulté/temps
- *Après* avoir commencé:
 - Commencer par les questions les plus faciles!
 - Bloqué? Sauter la question et prenez note d'y revenir
 - Ne pas oublier de respirer

Communication série

- Donnez la séquence de bits correspondant à l'envoi du mot 0x7A si on emploie la configuration suivante:
 - 8 bits par octet
 - parité paire
 - 2 bits d'arrêt

Communication série

- Donnez la séquence de bits correspondant à l'envoi du mot 0x7A si on emploie la configuration suivante:
 - 8 bits par octet
 - parité paire
 - 2 bits d'arrêt
- Séquence de bits: 001011110111
 - 0: bit de départ
 - 01011110: mot à envoyer (LSB en premier)
 - 1: parité paire
 - 11: bits d'arrêt

Ordonnancement de processus

- Les processus suivants sont admis, dans l'ordre, à l'ordonnancement de haut niveau

Processus	Durée	Temps d'arrivée	Priorité
P1	2	0	faible
P2	4	0	moyenne
P3	2	2	haute
P4	3	2	haute
P5	2	5	moyenne

- Indiquez quel processus sera exécuté à chaque quantum de temps si les algorithmes suivants sont utilisés:
 - Plus prioritaire d'abord. Si deux (ou plus) processus ont la même priorité, alors **premier arrivé, premier servi**
 - Plus prioritaire d'abord. Si deux (ou plus) processus ont la même priorité, alors **plus court d'abord**
 - Plus prioritaire d'abord. Si deux (ou plus) processus ont la même priorité, alors **tourniquet**

Ordonnancement de processus

- Plus prioritaire d'abord. Si deux (ou plus) processus ont la même priorité, alors **premier arrivé, premier servi**
 - P2-P2-P3-P3-P4-P4-P4-P2-P2-P5-P5-P1-P1
- Plus prioritaire d'abord. Si deux (ou plus) processus ont la même priorité, alors **plus court d'abord**
 - P2-P2-P3-P3-P4-P4-P4-P2-P2-P5-P5-P1-P1
- Plus prioritaire d'abord. Si deux (ou plus) processus ont la même priorité, alors **tourniquet**
 - P2-P2-P3-P4-P3-P4-P4-P2-P5-P2-P5-P1-P1

Mémoire paginée

- Dans un système avec mémoire paginée où:
 - les pages ont une taille de 2Ko;
 - le nombre maximal de pages est de 65 536;
 - la taille de la RAM est de 64Mo;
 - chaque élément de la table des pages contient le numéro de trame + 2 bits d'information supplémentaire

Extrait de la table des pages

# page	# trame
0	3C
1	8
2	367
3	F31
4	5D
5	41A
6	23
7	9A

- Quelle est la quantité maximale de mémoire virtuelle?

Mémoire paginée

- Dans un système avec mémoire paginée où:
 - les pages ont une taille de 2Ko;
 - le nombre maximal de pages est de 65 536;
 - la taille de la RAM est de 64Mo;
 - chaque élément de la table des pages contient le numéro de trame + 2 bits d'information supplémentaire

Extrait de la table des pages

# page	# trame
0	3C
1	8
2	367
3	F31
4	5D
5	41A
6	23
7	9A

- Quelle est la quantité maximale de mémoire virtuelle?
 - $65\ 536 \times 2\text{Ko} = \mathbf{128\ Mo}$

Mémoire paginée

- Dans un système avec mémoire paginée où:
 - les pages ont une taille de 2Ko;
 - le nombre maximal de pages est de 65 536;
 - la taille de la RAM est de 64Mo;
 - chaque élément de la table des pages contient le numéro de trame + 2 bits d'information supplémentaire
- Quelle est la taille de la table des pages?

Extrait de la table des pages

# page	# trame
0	3C
1	8
2	367
3	F31
4	5D
5	41A
6	23
7	9A

Mémoire paginée

- Dans un système avec mémoire paginée où:
 - les pages ont une taille de 2Ko;
 - le nombre maximal de pages est de 65 536;
 - la taille de la RAM est de 64Mo;
 - chaque élément de la table des pages contient le numéro de trame + 2 bits d'information supplémentaire
- Quelle est la taille de la table des pages?
 - taille = #entrées x taille d'une entrée
 - #entrées = #pages = 65 536
 - taille d'une entrée = $\log_2 \#trames + 2 \text{ bits}$
 - # trames = $64 \text{ Mo} / 2\text{Ko} = 2^{15}$
 - donc, taille de la table des pages = $65\,536 \times 17 \text{ bits} = \mathbf{136 \text{ Ko}}$

Extrait de la table des pages

# page	# trame
0	3C
1	8
2	367
3	F31
4	5D
5	41A
6	23
7	9A

Mémoire paginée

- Dans un système avec mémoire paginée où:
 - les pages ont une taille de 2Ko;
 - le nombre maximal de pages est de 65 536;
 - la taille de la RAM est de 64Mo;
 - chaque élément de la table des pages contient le numéro de trame + 2 bits d'information supplémentaire
- Traduisez l'adresse virtuelle 0x3B26 en adresse physique, et donnez sa valeur en hexadécimal

Extrait de la table des pages

# page	# trame
0	3C
1	8
2	367
3	F31
4	5D
5	41A
6	23
7	9A

Mémoire paginée

- Dans un système avec mémoire paginée où:
 - les pages ont une taille de 2Ko;
 - le nombre maximal de pages est de 65 536;
 - la taille de la RAM est de 64Mo;
 - chaque élément de la table des pages contient le numéro de trame + 2 bits d'information supplémentaire
- Traduisez l'adresse virtuelle 0x3B26 en adresse physique, et donnez sa valeur en hexadécimal
 - 0x3B26 = 0b0011 1011 0010 0110
(rouge = #de page, bleu = position dans la page)
 - pour la page 7, nous avons la trame 0x9A, soit 0b1001 1010
 - concaténons le # de trame avec la position dans la page:
0b100 1101 0011 0010 0110 = **0x4D326**
(vert = #de trame, bleu = position dans la page)

Extrait de la table des pages

# page	# trame
0	3C
1	8
2	367
3	F31
4	5D
5	41A
6	23
7	9A

Allocation mémoire contigüe

- Dans un système avec allocation mémoire contigüe avec partitions de taille *variable* et possédant une taille mémoire de 64Mo, les processus suivants sont alloués:
 - P1, 7Mo — P2, 2Mo — P3, 10Mo — P4, 8Mo — P5, 5Mo — P6, 2Mo
- Les processus P1, P3, et P5 se terminent
- Les processus suivants sont ensuite alloués:
 - P7, 5Mo — P8, 6Mo
- Écrivez le contenu de la mémoire si les algorithmes suivants sont utilisés pour allouer la mémoire:
 - Première allocation («first fit»)
 - Meilleure allocation («best fit»)
 - Pire allocation («worst fit»)

Allocation mémoire contigüe

- Dans un système avec allocation mémoire contigüe avec partitions de taille *variable* et possédant une taille mémoire de 64Mo, les processus suivants sont alloués:
 - P1, 7Mo — P2, 2Mo — P3, 10Mo — P4, 8Mo — P5, 5Mo — P6, 2Mo
- Les processus P1, P3, et P5 se terminent
- Les processus suivants sont ensuite alloués:
 - P7, 5Mo — P8, 6Mo
- Écrivez le contenu de la mémoire si les algorithmes suivants sont utilisés pour allouer la mémoire:

- **Première allocation («first fit»)**
- Meilleure allocation («best fit»)
- Pire allocation («worst fit»)

Adresses	Processus
0-5	P7
7-9	P2
9-15	P8
19-27	P4
32-34	P6

Allocation mémoire contigüe

- Dans un système avec allocation mémoire contigüe avec partitions de taille *variable* et possédant une taille mémoire de 64Mo, les processus suivants sont alloués:
 - P1, 7Mo — P2, 2Mo — P3, 10Mo — P4, 8Mo — P5, 5Mo — P6, 2Mo
- Les processus P1, P3, et P5 se terminent
- Les processus suivants sont ensuite alloués:
 - P7, 5Mo — P8, 6Mo
- Écrivez le contenu de la mémoire si les algorithmes suivants sont utilisés pour allouer la mémoire:

- Première allocation («first fit»)
- **Meilleure allocation («best fit»)**
- Pire allocation («worst fit»)

Adresses	Processus
0-6	P8
7-9	P2
19-27	P4
27-32	P7
32-34	P6

Allocation mémoire contigüe

- Dans un système avec allocation mémoire contigüe avec partitions de taille *variable* et possédant une taille mémoire de 64Mo, les processus suivants sont alloués:
 - P1, 7Mo — P2, 2Mo — P3, 10Mo — P4, 8Mo — P5, 5Mo — P6, 2Mo
- Les processus P1, P3, et P5 se terminent
- Les processus suivants sont ensuite alloués:
 - P7, 5Mo — P8, 6Mo
- Écrivez le contenu de la mémoire si les algorithmes suivants sont utilisés pour allouer la mémoire:

- Première allocation («first fit»)
- Meilleure allocation («best fit»)
- **Pire allocation («worst fit»)**

Adresses	Processus
7-9	P2
19-27	P4
32-34	P6
34-39	P7
39-45	P8