مجلس الخدمة المدنية اللجنة الفاحصة
مباراة لملء بعض المراكز الشاغرة في ملاك المديرية العامة للإستثمار في
وزارة الطاقة و المياه

لوظيفة : رئيس دائرة المعلوماتية
مسابقة في :أنظمة تشغيل الكمبيوتر الوقت: ساعتان

Exercice 1.

Paging technique:

- Explain the objective of paging in memory management;
- Explain address mapping mechanism. What is the used hardware support and how is it used?
- What happens if a virtual address is not loaded into physical memory. List three swap strategies.
- What is the difference between one-level and two-level paging?
- Given processor with address space of 4 GB and physical memory of 128 MB divided into 4096 page frames. Determine the page size, the number of maximum virtual pages, and the format of the virtual address.

Exercice 2.

Does the following solution to the critical section problem for two processes satisfy the desired mutual exclusion requirements (mutual exclusion, no deadlock, no starvation, process can enter empty critical section [do nothing] without delay)? Justify your response by discussing each of the critical section requirements, if the solution is correct; or one of the critical section requirements, if the solution is not correct.
We have two flags (shared variables): Cole and Lope, also we have neon SIGN (shared variable) which flashes either "Nicole" or "Penelope".
Initially: both flags are down and SIGN flashes "Nicole". The operation "raise" on a shared variable changes its state to up, while the operation "lower" changes the state to down.

Exercise 1.

Technique de pagination

- Expliquer l'objectif des mécanismes de pagination dans la gestion de mémoires:
- Expliquer le mécanisme de traduction des adresses et la fonction de topographie; Quel est le support matériel utilisé pour cette traduction et comment est-il utilisé?
- Qu'est-ce qui se passe si une adresse virtuelle n'est pas chargée en mémoire physique. Citez trois stratégies de va-et-vient.
- Quelle est la différence entre pagination à un niveau et pagination à deux niveaux?
- Soit un processeur d'espace d'adressage 4 Go et une mémoire physique de 128 Mo, divisé en 4096 pages physiques. Déterminer la taille de la page, le nombre maximum de pages virtuelles, et le format de l'adresse virtuelle.

Exercise 2.

Est-ce que la solution suivante d'un problème de section critique pour deux processus répond aux exigences souhaitées d'une exclusion mutuelle (exclusion mutuelle, aucun interblocage, pas de famine, le processus peut entrer dans une section critique vides [do nothing] sans retard)? Justifiez votre réponse en discutant chacune des exigences de la section critique pour conclure si la solution est correcte, ou l'une des exigences d'une section critique est correcte ou si la solution n'est pas correcte.
Nous avons deux drapeaux (variables partagées): Cole et Lope et encore un néon SIGNAL (variable partagée) qui clignote soit "Nicole" soit "Pénélope".
D'abord: les deux drapeaux sont en baisse et le signal clignote "Nicole".

Nicole	Penelope

repeat
[play with kids];
raise Cole;
while (Lope is up) {
if (SIGN == Penelope) {
lower Cole;
while (SIGN == Penelope)
[do nothing];
raise Cole;
}
}
[eat dinner];
SIGN = Penelope
lower Cole;
forever;
	repeat
	[call Tom];
	raise Lope;
	while (Cole is up) {
	if (SIGN == Nicole) {
	lower Lope;
	while (SIGN == Nicole)
	[do nothing];
	raise Lope;
	}
	}
	[eat dinner];
	SIGN = Nicole
	lower Lope;
	forever;

Exercice 3.

During a process ***switch***, the operating system executes instructions that choose the next process to execute. These system instructions are typically at a fixed location in memory. Why?

Exercice 4.

Consider the following pseudo assembly code for computing c = a + b. Assume that a, b, and c are assigned to consecutive memory "words" (memory is generally addressed byte by byte and assume that a word is 4 bytes) and address for "a" is 0x0000EC00. Also, we have a = 22, b = 158, and c = 0 at the starting time. Assume that the first instruction of the code is stored in 0x0000B128. Also, each instruction has the OpCode in the first byte (most significant byte) and the remaining 3 bytes specify the corresponding address. The OpCode for store is 1, load is 2, and add is 3.

0x0000b128 load a
0x0000b12c add b
0x0000b130 store c

Show the memory addresses and contents for all the instructions and data involved. Use the format as follows to express your answer (but the following is not the answer). For all data, use hexadecimal representation.

addresses	**contents**
0x00002104	0x00000001
0x00002108	0x00000002

Nicole	Penelope

repeat
[play with kids];
raise Cole;
while (Lope is up) {
if (SIGN == Penelope) {
lower Cole;
while (SIGN == Penelope)
[do nothing];
raise Cole;
}
}
[eat dinner];
SIGN = Penelope
lower Cole;
forever;
	repeat
	[call Tom];
	raise Lope;
	while (Cole is up) {
	if (SIGN == Nicole) {
	lower Lope;
	while (SIGN == Nicole)
	[do nothing];
	raise Lope;
	}
	}
	[eat dinner];
	SIGN = Nicole
	lower Lope;
	forever;

Exercise 3.

Lors du processus switch, le système d'exploitation exécute les instructions qui choisissent le prochain processus à exécuter. Ces instructions système sont généralement à un emplacement fixe dans la mémoire. Pourquoi?

Exercise 4.

Considérons le pseudo code assembleur suivant pour le calcul de c = a + b. Supposons que a, b et c sont affectés à des "mots" consécutifs d'une mémoire séquentielle (La mémoire est généralement adressée octet par octet et le mot soit de 4 octets) et l'adresse de "a" est 0x0000EC00. De plus, nous avons a = 22, b =158, et c = 0 au moment de départ. Supposons que la première instruction du code est stockée dans 0x0000B128. . En outre, chaque instruction a l' OpCode dans le premier octet (octet le plus significatif) et les 3 octets restants spécifient l'adresse correspondante. L'OpCode pour la stocke est 1, pour le chargement est 2, et pour l'addition est 3.

0x0000b128 load a
0x0000b12c add b
0x0000b130 store c

Présenter les adresses mémoire et les contenus pour toutes les instructions et les données impliqués. Utilisez le format suivant pour exprimer votre réponse (mais ce qui suit n'est pas la réponse). Pour toutes les données, utiliser la représentation hexadécimale.

adresses	contenus
0x00002104	0x00000001
0x00002108	0x00000002

بيروت في 2011/8/9

اللجنة الفاحصة