

IN1805-I Operating System Concepten

23 juni 2010, 09.00 - 12.00 uur.

docent: D.H.J. Epema

Dit is een tentamen met 8 open vragen

Opgave	Punten
1	12
2	12
3	12
4	14
5	12
6	14
7	12
8	12
Totaal	100

Het gebruik van boeken, dictaten of aantekeningen is **niet** toegestaan

Schrijf **duidelijk** en formuleer **kort en bondig**

Opgave 1 (12 punten)

Veel operating systemen kennen een *user mode* en een *kernel mode*.

- Leg uit wat deze begrippen inhouden.
- Wat is het belangrijkste probleem in een systeem dat maar één mode kent? Geef een korte toelichting.
- Hoe (en wanneer) gaat een systeem van user mode naar kernel mode? En wanneer gaat een systeem van kernel mode naar user mode? Geef een korte toelichting.

Opgave 2 (12 punten)

Gegeven de volgende drie toestanden waarin een proces kan zich bevinden: *Ready*, *Running*, en *Waiting*.

- Beschrijf voor iedere toestand kort wat het betekent dat een proces in die toestand is.
- Geef in een eenvoudig toestandsdiagram door middel van pijlen aan welke toestandsovergangen tussen deze toestanden mogelijk zijn. Geef in korte bewoordingen aan wanneer deze overgangen optreden.
- Beschrijf kort wat wordt verstaan onder een Process Control Block (PCB).
- Is in het PCB van een running proces op ieder moment de waarde van de Program Counter te vinden? Motiveer uw antwoord.

Opgave 3 (12 punten)

Voor synchronisatie van processen wordt vaak gebruik gemaakt van *semaforen*.

- Leg uit wat er in een operating system wordt verstaan onder een semafoor.
- Welke operaties kunnen op semaforen worden uitgevoerd? Beschrijf kort de werking van deze operaties.
- Gegeven twee processen A en B die om de beurt een *shared variable* x wijzigen. Proces A geeft x een waarde, proces B voert hiermee een berekening uit en geeft x een nieuwe waarde, hierna gebruikt A deze waarde en berekent een nieuwe waarde, etc.

Beschrijf hoe de gewenste synchronisatie tussen A en B kan worden geregeld met behulp van semaforen:

- Geef hierbij aan hoeveel semaforen er worden gebruikt (geef ze een naam).
- Geef aan wat de initiële waarden zijn.
- Geef aan waar in A en B welke operaties op de semaforen worden uitgevoerd.

Opgave 4 (14 punten)

Een begrip dat een rol speelt bij het ontwijken (*avoidance*) van deadlocks is *veilige volgorde*.

- a. Leg uit wat er verstaan wordt onder een veilige volgorde.
- b. Stel we hebben een systeem met 8 buffers. Onderstaande tabel geeft de allocatie van deze buffers aan de processen A, B, en C op een bepaald moment weer, en geeft ook de maximale behoefte van de processen aan.

proces	allocatie	maximum
A	1	4
B	2	3
C	3	7

Laat zien dat dit een veilige toestand is.

- c. Proces C vraagt nu zijn vierde buffer aan. Mag deze worden toegewezen als we deadlocks willen ontwijken? Licht uw antwoord kort toe.
- d. Als een toewijzing niet zou mogen plaatsvinden en het zou toch gebeuren, gaat er dan gegarandeerd een deadlock optreden? Motiveer uw antwoord.

Opgave 5 (12 punten)

In een hoofdgeheugen wordt van *contiguous allocation* gebruik gemaakt. Op een bepaald moment zijn er vier geheugenpartities M_1 t/m M_4 vrij van afmeting 100 KB, 500 KB, 200 KB, resp. 300 KB (in deze volgorde in het geheugen). Er moeten vier processen P_1 t/m P_4 in het geheugen worden geplaatst met afmetingen 212 KB, 417 KB, 112 KB, resp. 426 KB (eveneens in deze volgorde). Geef aan waar de processen in het geheugen worden geplaatst bij gebruik van de volgende allocatie-policies:

- a. First Fit
b. Best Fit
c. Worst Fit

Opgave 6 (14 punten)

- a. Wat wordt er verstaan onder *demand paging*?
- b. Gegeven de volgende *Page Reference String*:

0 1 2 0 3 1 0 1 2 3 1 3 1 0

Neem aan dat er drie *page frames* beschikbaar zijn, genummerd 0, 1 en 2. Geef voor ieder van de volgende *page-replacement* algoritmen in de vorm van een tabel weer, hoe na iedere referentie de verdeling van de pages over de frames is:

- First-In, First-Out (FIFO)
- Least Recently Used (LRU)
- MIN (of OPT)

Bepaal steeds ook het aantal pagefaults.

- c. Is het aantal pagefaults bij de verschillende algoritmen onder b.) overeenkomstig hetgeen we zouden mogen verwachten? Licht uw antwoord kort toe.

Opgave 7 (12 punten)

Operating systemen houden vaak zowel *per-process file tables* (PFT) bij als een *system-wide file table* (SFT).

- a. Om welke reden houden operating systemen beide soorten file tables bij?
- b. Geef van vier typen gegevens m.b.t. (het gebruik van) files aan of deze typisch in de PFTs worden opgeslagen of in de SFT.
- c. Een entry in de SFT kan voor verschillende processen in gebruik zijn. Op welke manier kan het operating system bepalen of een entry in de SFT kan worden verwijderd als een proces de bijbehorende file sluit?

Opgave 8 (12 punten)

Het Access Matrix model is een concept dat gebruikt wordt om de protectieregels bij te houden die het operating system af moet dwingen.

- a. Licht de begrippen *object*, *protectiedomein* en *access recht*, en hun onderlinge relatie, kort toe.
- b. Leg uit waarom het bijhouden van de Access Matrix praktisch gezien niet haalbaar is, en beschrijf de twee meest gebruikte implementatietechnieken: *access lists* en *capabilities*.
- c. Beschrijf hoe capabilities m.b.v. *one-way functions* geïmplementeerd (en geverifieerd) kunnen worden.

Einde tentamenopgaven