

## Eerste deeltentamen Netwerken (INFONW) 18 mei 2009

Open vragen (8 vragen): 60% van het cijfer.

Geef bij alle open vragen een duidelijke uitleg of motivering van het antwoord.

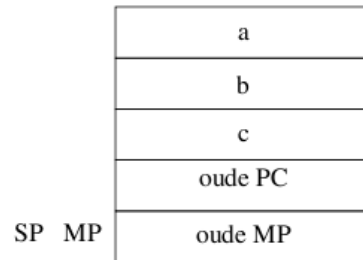
### Opgave 1

Stel we hebben een geheugen met een toegangstijd van 20 nsec. Dat wil zeggen dat het 20 nsec tijd kost om een woord uit het geheugen te halen en in het cache te zetten ( $1 \text{ nsec} = 10^{-9} \text{ sec}$ ). De communicatie tussen de CPU en het cache kost 4 nsec per woord. De hit rate van het cache is 80%, dat wil zeggen dat 80% van elk opgevraagd woord in het cache te vinden is. Hoeveel tijd kost het gemiddeld om een woord uit het geheugen in de CPU te krijgen?

### Opgave 2

In SSM hebben we op een gegeven moment de stack zoals in het plaatje rechts. We gaan daarna de volgende instructies uitvoeren:

```
ldl -3
lds 0
mul
ldl -2
lds 0
mul
sub
stl -4
```



Geef bij elke instructie in bovenstaand programma precies aan wat die doet of berekent. Wat is er na afloop veranderd in het plaatje?

Hier is een overzicht van de gebruikte instructies:

instructie	pop	push	operatie
add	2	1	$a + b$
sub	2	1	$a - b$
mul	2	1	$a \times b$
lds $n$	0	1	load woord $n$ t.o.v. SP
ldl $n$	0	1	load woord $n$ t.o.v. MP
stl $n$	1	0	store woord $n$ t.o.v. MP

*NB. hierbij is  $b = \text{de top van de stack}$ ;  $a = \text{het volgende woord}$*

### Opgave 3

Stel we willen in een file een (objectgeïntereerde) database maken, die bestaat uit een grote verzameling objecten die onderling naar elkaar verwijzen.

1. Leg uit waarom het gebruik van een memory-mapped file hiervoor handiger is dan het gebruik van `read` en `write` methodes.
2. In een programma willen we enkele van de objecten uit deze database bekijken en/of wijzigen. We gebruiken hiervoor de memory-mapped file. Heeft dit tot gevolg dat de hele file in het fysieke geheugen geladen wordt? **Beargumenteer het antwoord.**

### Opgave 4

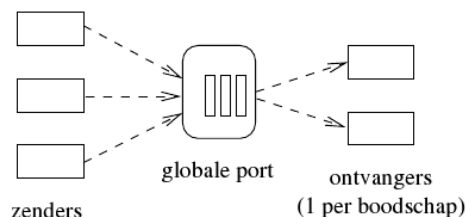
1. Leg uit wat een toestandsloze (stateless) server is.
2. Waarom is file locking in een toestandsloze server moeilijk te realiseren?

### Opgave 5

Beschrijf het verschil tussen processen en threads.

### Opgave 6

Onderstaande plaatje van een globale port zou ook gebruikt kunnen worden voor een broadcast port. Leg uit wat het verschil tussen beide is.



### Opgave 7

1. Is een multithreaded server altijd efficiënter dan een singlethreaded server?
2. Als het antwoord op (a) ja is, leg dan uit waarom dit zo is. Als het antwoord op (a) nee is geef dan een voorbeeld waarbij het wel efficiënter is en een voorbeeld waarbij het niet zo is.

### Opgave 8

Beschrijf twee belangrijk verschillen tussen het gebruik van *pipes* en *message passing* als IPC mechanisme.

Multiple choicevragen (10 vragen): 40% van het cijfer. Bij deze vragen is telkens één antwoord goed. Wanneer een antwoord op zich goed is, maar een ander beter, moet je dat laatste kiezen.

### **Opgave 1**

We kunnen zeggen dat een computer een 64-bits systeem is als:

1. er 64 registers in de CPU zijn.
2. de bus 64 bits tegelijk kan verwerken.
3. de adressen en registers 64 bits groot zijn.
4. elk van bovenstaande

### **Opgave 2**

Welke gebeurtenis veroorzaakt een trap?

1. Het bewegen van de muis.
2. Delen door 0.
3. Zowel a als b.
4. Geen van bovenstaande.

### **Opgave 3**

Wat is het verschil tussen segmentering en paginering?

1. Pagina's hoeven niet in het fysieke geheugen te staan.
2. Alle segmenten moeten even groot zijn.
3. Pagina's kunnen verschillende lengtes hebben.

### **Opgave 4**

Wat gebruikt een modern O.S. om het geheugen van een proces te beschermen tegen andere processen?

1. scheduling
2. paginering
3. threads
4. file locking

### **Opgave 5**

Copy-on-write is een techniek om:

1. met behulp van memory management hardware kopiëren uit te stellen.
2. het disk cache te kopiëren naar de harde schijf.
3. een link naar een file te maken in plaats van een kopie.
4. een nieuw proces te starten.

## Opgave 6

Wat is de beste definitie van een *cache*?

1. Een snel stuk geheugen tussen de CPU en de bus.
2. Een stuk geheugen dat gebruikt wordt om pagina's van de harde schijf op te slaan.
3. Een stuk geheugen/schijf dat gebruikt wordt om documenten uit het WWW op te slaan.
4. Een stuk snelle opslagruimte dat gebruikt wordt om een deel van een langzaam opslagmedium op te slaan.

## Opgave 7

Gebufferde I/O wordt gebruikt om:

1. minder systeemaanroepen te hoeven doen.
2. het disk-cache beter te benutten.
3. files niet in het geheugen te hoeven laden.
4. files te sharen tussen verschillende processen.

## Opgave 8

Bij een *context switch* wordt:

1. een interrupt verwerkt
2. de pagetable veranderd
3. een nieuw proces gecreëerd
4. een nieuwe thread gecreëerd

## Opgave 9

Het doel van file locking is:

1. het beschermen van files tegen ongeoorloofd gebruik door andere gebruikers.
2. het beschermen van files die tot het operating system behoren.
3. het beschermen van files tegen gelijktijdig gebruik door verschillende processen en/of threads.
4. het vastleggen van een file in het geheugen.

## Opgave 10

Bij Remote Procedure Call is een *stub* nodig die de marshalling doet omdat:

1. de manier waarop getallen opgeslagen worden anders kan zijn bij beide computers.
2. pointers speciaal behandeld moeten worden bij RPC.
3. parameters die door de procedure veranderd worden speciaal behandeld moeten worden.
4. a, b, en c.