

MEMORANDUM

Universiteit van Stellenbosch Departement Rekenaarwetenskap

Kursus: RW 314

Eksamen, Junie 2002 Tydsduur: 3 uur

Instruksies *Instructions*

- (a) Skryf netjies en met ink.
Write neatly in ink.
- (b) Sakrekenaars word toegelaat.
Calculators may be used.
- (c) Toon alle berekenings indien van u verlang.
Show all calculations if required.

Waar of Onwaar *True or False*

1. Beantwoord die volgende vrae deur aan te dui of elke bewering WAAR of ONWAAR is.

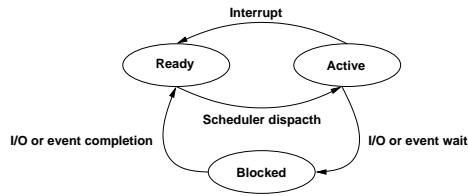
Answer the following questions by indicating whether each statement is TRUE or FALSE.

- (a) Kondisieveranderlikes word gebruik om onderlinge uitsluiting te waarborg in stelsels wat gebaseer is op monitors.
Condition variables are used to guarantee mutual exclusion in systems that are based on monitors. **T**
- (b) 'n Kritiese gebied is 'n datastruktuur wat deur meer as een proses bygewerk kan word.
A critical region is a data structure that can be updated by more than one process. **F**
- (c) Vergrendeling is onmoontlik indien slegs een hulpbron op 'n slag aan elke proses toegeken kan word.
Deadlock is impossible if only one resource can be assigned to each process at a time. **F**
- (d) Dit is onmoontlik dat meer as een bladyfout kan gebeur tydens uitvoering van 'n enkele instruksie.
It is impossible for more than one page fault to occur during execution of a single instruction. **F**
- (e) Die toekenning van bladsye aan prosesse kan beheer word deur die frekwensie van bladsyfoute te monitor.
The allocation of pages to processes can be controlled by monitoring the frequency of page faults. **T**
- (f) Mutlivlak terugvoertou skedulering is regverdig.
Multilevel feedback queue scheduling is fair. **F**

12

2. 'n Proses se toestand kan verander tussen “gereed”, “aktief”, en “geblokkeer”. Teken 'n diagram om aan te toon watter toestandsveranderinge moontlik is en watter gebeurtenis elke toestandsverandering sal veroorsaak.

The state of a process can change between “ready”, “active”, and “blocked”. Draw a diagram to show which state changes are possible and what event will cause each state change.



6

3. Noem en bespreek twee vereistes vir vergrendeling.

Name and discuss two requirements for deadlock.

- (a) **Mutual Exclusion** - Resources must be nonsharable so that only one process can obtain the resource while other processes must wait until the resource is released.
- (b) **Hold and Wait** - A process must hold a resource while waiting to allocate additional resources held by other processes.

6

4. Verduidelik wat bedoel word met die term **direkte geheue-toegang** (DMA), met spesiale verwysing na die gebruik daarvan in skyf-beheereenhede en hoe data oorgedra word tussen programmatuur en die skyf-beheereenheid.

*Explain what is meant by the term **direct memory access** (DMA) with special reference to its use in disk control units and how data is transferred between software and the disk controller.*

DMA refers to a mechanism that supports asynchronous I/O. When the operating system receives a request to transfer data to or from an I/O device, it initializes the DMA controller by sending various parameters such as the amount of data and the address of the buffer that must be used during the transfer. Once the DMA controller has been programmed, a command is sent to the I/O device to either read or write the data using DMA. Once the command has been sent to the device, the operating system can continue by scheduling another process. The I/O device and DMA controller will then exchange the data between themselves without the requiring any intervention from the CPU. When the transfer is complete, the I/O device will generate an interrupt, signalling that the operation has been completed. Although DMA allows the CPU to continue with other processing operations, some CPU cycles are lost (cycle stealing) because there will be contention between the CPU and DMA controller to access the system bus. If the I/O requested access to a disk, the operating system will begin by initializing the DMA controller, before sending either a read or write command to the disk controller. Once the disk controller receives the command, it will begin executing the required operation and exchange the data with the DMA controller until the operation has been completed after which the disk controller will generate an interrupt and control is transferred back to the operating system.

6

5. Bestudeer die onderstaande implementasie van die “Dining Philosopher’s” probleem en beantwoord dan die vrae wat volg.

Examine the implementation of the Dining Philosopher’s problem below and answer the questions that follow.

```

VAR
  Forks: ARRAY 5 OF Semaphore;
...
(* Implementation of a single philosopher *)
PROCESS Philosopher(i: INTEGER);
BEGIN
  WHILE TRUE DO
    IF i MOD 2 = 0 THEN
      Wait(Forks[i]);
      Wait(Forks[(i+1) MOD 5]);
      (* Eat... *)
      Signal(Forks[i]);
      Signal(Forks[(i+1) MOD 5]);
      (* Think... *)
    ELSE
      Wait(Forks[(i+1) MOD 5]);
      Wait(Forks[i]);
      (* Eat... *)
      Signal(Forks[(i+1) MOD 5]);
      Signal(Forks[i]);
      (* Think... *)
    END;
  END
END Philosopher

BEGIN
  Create(Philosopher(0), Philosopher(1), Philosopher(2),
        Philosopher(3), Philosopher(4))
END.

```

- (a) Kan vergrendeling voorkom? Motiveer u antwoord.

Does the implementation contain a deadlock? Motivate your answer.

No. This is an asymmetrical solution that guarantees that the circular wait condition never prevails. Either even numbered ($i \bmod 2 = 0$) or odd numbered ($i \bmod 2 \neq 0$) philosophers may eat. For example, if philosopher 1 obtains his/her fork ($\text{Forks}[(i+1) \bmod 5]$), then philosophers 2 will not be able to obtain the same fork because it will attempt execute $\text{Wait}(\text{Forks}[i])$.

5

- (b) Is dit moontlik dat die verhongering van proses sal voorkom? Motiveer u antwoord.

Can the implementation result in the starvation of a philosopher? Motivate your answer.

Yes. The solution does not provide for a philosopher to postpone eating again once he/she has done so already (there is no fairness). It is possible that the same philosopher can eat, put down his/her forks and then eat again before the adjacent philosophers get a chance to eat.

3

6. Gee **twee** goeie redes waarom die “banker’s” algoritme (vir die voorkoming van vergrendeling) nie baie bruikbaar is in die praktyk nie.

Give two good reasons why the banker’s algorithm (for the prevention of deadlock) is not very useful in practice.

- (a) The banker's algorithm requires that every process states the maximum number of every resource that it would require. This information is often unknown in practice.
- (b) Resources, especially hardware, may break down from time to time. However, the banker's algorithm requires that the number of resources stay constant.

2

7. Verduidelik kortliks die verskil tussen interne en eksterne fragmentasie ten opsigte van primêre geheue.

Briefly explain the difference between internal and external fragmentation with regards to primary memory.

Gaps are created in memory as processes are loaded and removed from it. Eventually the memory is broken into tiny fragments that in total can satisfy the memory requirements of a process, but is not contiguous. This is called external fragmentation. Internal fragmentation occurs when a block of memory is allocated, but the difference between the available block and the amount requested is too small to warrant keeping track of it. Instead, the operating system will allocate the whole block.

3

8. Beskryf kortliks wat gebeur wanneer 'n proses 'n bladsyfout genereer.

Briefly describe what happens when a process generates a page fault.

Whenever a page fault occurs, the processor will trap to the operating system. The operating system is then free to determine which process is responsible for the fault and determines if the memory reference was valid. If not, the process is terminated. If the reference was valid, the page must be brought into main memory. First, a free frame is allocated and a disk operation is scheduled to retrieve the contents of the page from secondary storage. Once the page has been loaded, the operating system creates an entry in the page table of the process before restarting the instruction that caused the page fault.

6

9. Verduidelik kortliks die begrip van adresbinding en bespreek die tegniek van looptydbinding met spesifieke verwysing na die vereistes wat gestel word om dit suksesvol te ondersteun.

Briefly describe the concept of address binding and discuss the use of execution (runtime) binding with specific reference to the requirements that must be met to successfully support it.

Address binding allows addresses to be mapped between different address spaces. Binding during runtime allows an operating system to move a process within the address space without compromising the addresses generated by the process. Special hardware is necessary to support this scheme, because the logical and physical addresses differ. A *relocation* register is used as a base address and its value is added to every address generated by a process. This is done by the memory management unit (MMU).

8

10. Definieër die term sparteling. Bespreek kortliks hoe die **Werkverwysingsmodel** gebruik kan word om hierdie probleem aan te spreek.

*Define the term thrashing and briefly describe the application of the **Working-Set Model** as a possible solution to this problem.*

Thrashing occurs when a process spends more time paging than executing. The working set model is an approximation of a process' locality. The working set window, δ , defines the size of the working set. Once loaded, pages will stay in the working set for at least δ references before being dropped after its last reference. This strategy prevents thrashing and attempts to increase the degree of multiprogramming and CPU utilization.

5

11. Verduidelik kortliks die verskil tussen laevlak en logiese formatering.

Briefly describe the difference between lowlevel and logical formatting.

Low-level formatting initializes the physical structure of a disk by dividing it into sectors and initializes every sector with certain information such as the sector number and ECC values. Logical formatting creates the file system by initializing specific sectors with certain structures such as a FAT or i-node table.

4

12. 'n Sekere lêerstelsel ken skyfblokke toe deur middel van die indekstegniek. Elke lêer het 'n indeksnodus wat 64 grepe beslaan. Die eerste 24 grepe in die indeks word gebruik om lêereienskappe te stoor. Die res van die indeks word gebruik om die skyfadresse (4 grepe per adres) van datablokke te stoor. Die laaste twee skyfadresse in die indeksnodus word gebruik om indirekte en dubbelindirekte skyfadressering te ondersteun. Skyfblokke beslaan elk 512 grepe. Beantwoord die volgende vrae in verband met hierdie ontwerp.

A given file system allocates disk blocks by using the index technique. Each file has an index block that occupies 64 bytes. The first 24 bytes in the index are used to store file attributes. The rest of the index is used to store disk addresses (4 bytes per address) of data blocks. The last two block entries in index node are used to support indirect and double indirect disk addressing. Disk blocks occupy 512 bytes each. Answer the following questions regarding this design.

- (a) Wat is die grootte van die grootste lêer wat gestoor kan word sonder om van indirekte en dubbelindirekte adressering gebruik te maak?

What is the size of the largest file that can be stored without using indirect or double indirect addressing?

| $8 \times 512 = 4096$ bytes

2

- (b) Wat is die grootte van die grootste lêer wat gestoor kan word?

What is the size of the largest file that can be stored?

- For single indirect: $\frac{512}{4} \times 512 = 65536$
- For double indirect: $\frac{512}{4} \times 65536 = 8388608$
- Maximum file size: $4096 + 65536 + 8388608 = 8458240$ bytes

4

- (c) Beskryf kortliks een metode om die reaksietempo van die lêerstelsel te verbeter.

Describe briefly one method to improve the reaction time of the file system.

The index nodes of file can be kept in memory and will reduce the number of disk accesses required because the data blocks can be accessed directly without having to first read the index node.

4

- (d) Noem 'n belangrike rede waarom die blokke van dieselfde lêer gewoonlik nie in opeenvolgende posisies gestoor word op skyf nie.

Mention an important reason why blocks of the same file are usually not stored in consecutive positions on the disk.

Any free block can satisfy a request for more space. Furthermore, index nodes support both sequential and direct access methods while contiguous allocation can not support direct access efficiently.

2

13. Kontensie is dikwels 'n probleem wat in meeste netwerke ondervind word. Bespreek kortliks die oorsake en moontlike oplossings vir hierdie probleem.

Most networks suffer from contention. Briefly describe the reasons for this problem and discuss the various solutions.

It is possible that more than one link exist between two nodes in a network. In this case, it becomes possible that various nodes may attempt to transmit information over the same link simultaneously. A number of solutions exist:

- (a) Carrier Sense with Multiple Access and Collision Detection (CSMA/CD): Before transmitting, a node must listen to determine whether another node is currently using the link. A collision detection occurs when two nodes transmit information at the same time.
- (b) Token passing (ring): A special message called a token is passed from node to node in the network. Once the token arrives at a particular node, the token is removed and the node begins transmitting its message. Once the transmission has been completed, the token is retransmitted.
- (c) Message slots: A fixed number of message slots are circulated. A node may either place a message into an empty slot, or remove a message that was sent by another node to it.

14. Verskaf kort beskrywings vir die volgende:

Briefly describe the following:

(a) Virtuele Roetering

Virtual Routing

A path between two nodes is fixed for only one session and multiple messages between these two nodes may travel along different paths.

2

(b) Roeteerder

Router

A router can either be implemented as dedicated hardware or a host computer connected to at least two networks. Messages received by the router are examined and forwarded to other networks or hosts depending on its internet address.

2

(c) Kanaalskakeling

Circuit Switching

A physical connection between two entities used for the duration of communication. No other entity can use the link while it is open.

2

(d) TCP

TCP

Transmission Control Protocol. TCP together with the internet protocol (IP) forms the basis for transmitting packets of data between nodes in a network.

2

Totaal: / *Total:* 90