Afdeling Informatica	**Bedrijfssystemen**
Vrije Universiteit	**04.06.2008**

1a Een bedrijfssysteem kan gezien worden als een virtuele machine of als een resource manager. Leg het verschil uit. *5pt*

1b Leg het verschil uit tussen de *raw*, *cooked*, en *cbreak* terminal modes. *5pt*

2a Geef een uitleg van de *flow of control* als HW interrupts binnenkomen. *10pt*

2b Wat is DMA? Hoe werkt het, en wat zijn de voordelen/nadelen? *5pt*

3a Wat is het verschil tussen processen en threads? Welke informatie wordt respectievelijk opgeslagen? *5pt*

3b Wat zijn de voordelen/nadelen van thread implementatie in user vs. kernel space? *5pt*

4a Wat is de doel van de TSL instructie? Leg de voordelen uit van deze HW-gebaseerde oplossing, in tegenstelling tot andere SW-gebaseerde oplossingen. *5pt*

4b Geef een oplossing voor de Dining Philosophers probleem dat deadlock en starvation vermijdt, en dat parallelisatie maximaliseert. *10pt*

4c Beschouw de onderstaande toekenning van resources R1 t/m R4. Laat zien dat dit een *safe state* is. *5pt*

Process	R1	R2	R3	R4
A	4	1	0	1
B	0	2	0	0
C	1	0	1	0
D	1	0	0	1
E	0	0	1	0

Allocated resources

Process	R1	R2	R3	R4
A	1	1	0	0
B	0	1	1	2
C	3	1	0	0
D	0	0	1	1
E	2	1	1	0

Resources still needed

R1	R2	R3	R4
7	4	2	2

Originally available

5a Leg 3 Disk Arm Scheduling algorithmen uit. *5pt*

5b Welke type disk arm scheduling gebruikt MINIX3? Leg uit waarom. *5pt*

6a Leg uit de hoofdkenmerken van het a.out executable bestandsformat *5pt*

6b Beschouw een standaard organisatie van een UNIX filesysteem. Het bijhouden van beschikbare blokken of schijf en beschikbare i-nodes gebeurt d.m.v. bit maps. Hoe bepaal je het maximaal aantal bestanden dat een filesysteem aankan? *5pt*

6c Beschouw wederom een UNIX filesysteem. Hoe bepaal je de maximale grootte van een bestand? *5pt*

7a Wat is een *buffer cache* en waarom is het nuttig? Geef ook een korte uitleg over de block write-through strategieën van MS-DOS vs. UNIX. *5pt*

7b In MINIX3, wat doet de procedure DO_RDWT precies (zie de code op de andere pagina)? Leg elke stap uit, en geef de context aan waarin de procedure gebruikt wordt. *5pt*

```
11148  PRIVATE int do_rdwt(dp, mp)
11149  struct driver *dp;
11150  message *mp;
11151  {
11153    iovec_t iovec1;
11154    int r, opcode;
11155    phys_bytes phys_addr;
11156
11158    if (mp->COUNT < 0) return(EINVAL);
11159
11161    sys_umap(mp->PROC_NR, D, (vir_bytes) mp->ADDRESS, mp->COUNT, &phys_addr);
11162    if (phys_addr == 0) return(EFAULT);
11163
11165    if ((*dp->dr_prepare)(mp->DEVICE) == NIL_DEV) return(ENXIO);
11166
11168    opcode = mp->m_type == DEV_READ ? DEV_GATHER :  DEV_SCATTER;
11169    iovec1.iov_addr = (vir_bytes) mp->ADDRESS;
11170    iovec1.iov_size = mp->COUNT;
11171
11173    r = (*dp->dr_transfer)(mp->PROC_NR, opcode, mp->POSITION, &iovec1, 1);
11174
11176    return(r == OK ? (mp->COUNT - iovec1.iov_size) :  r);
11177  }
```

Cijferbepaling: *Het cijfer wordt bepaald door de punten die per onderdeel behaald zijn, bij elkaar op te tellen (totaal maximaal 90 punten), en vervolgens daar 10 extra punten bij te tellen. Er zijn dus totaal 100 punten te behalen.*