
Alle vragen tellen even zwaar. Het eindcijfer is de som van de deelcijfers.
For questions in English, please turn over.

1. Een UNIX program begint als volgt:

```
main() {  
  int fd1, fd2;  
  char buf[1000];  
  close(0);  
  close(1);  
  close(2);  
  fd1 = open("abc", 0);  
  fd2 = open("rst", 0);  
  fd3 = open("xyz", 0);  
  close(1);  
  dup(2);  
  read(1, buf, 100);  
}
```



Wordt iets gelezen? Zo ja, van welke file? Ga er vanuit dat abc, rst, en xyz allemaal bestaan, leesbaar zijn, en langer dan 100 bytes zijn. Bovendien, open geeft altijd de laagst beschikbare file descriptor terug.

2. Een monitor heeft twee procedures, 'producer' en 'consumer'. Is het mogelijk dat een proces 'producer' uitvoert en een tweede proces 'consumer' uitvoert en dat beide zich tegelijk binnen de monitor zijn? HEEL ERG GOED nadenken en daarna je antwoord toelichten.
3. Hieronder is een stukje code van de MINIX scheduler. Stel je voor dat MINIX processen zouden verdeeld worden in hoog en laag prioriteitsklassen. Geef de code voor de nieuwe scheduler.

```
PRIVATE void pick_proc()  
{  
  register struct proc *rp;  
  
  if ( (rp = rdy_head[TASK_Q]) != NIL_PROC) {  
    proc_ptr = rp;  
    return;  
  }  
  if ( (rp = rdy_head[SERVER_Q]) != NIL_PROC) {  
    proc_ptr = rp;  
    return;  
  }  
  if ( (rp = rdy_head[USER_Q]) != NIL_PROC) {  
    proc_ptr = rp;  
    bill_ptr = rp;  
    return;  
  }  
  bill_ptr = proc_ptr = proc_addr(IDLE);  
}
```

4. Leg het essentiële verschil uit tussen 'deadlock avoidance' (b.v. bankiers algoritme) en deadlock prevention (b.v. geordende resources).
5. Een MINIX programma lees van standard input (het toetsenbord) met de systeemaanroep `n = read(fd, buf, 10)`. De gebruiker typt:
d s <backspace> a t e <enter>
waar <backspace> en <enter> de bijbehorende toetsen zijn.
- (a) Wat is de waarde van n in 'cooked' mode?
(b) Wat is de waarde van n in 'raw' mode?