

### Aufgabe 1: Dateisysteme (4 Punkte)

Wieviel MB beträgt die maximal unterstützte Partitionsgröße eines FAT16 Filesystemes bei einer Blockgröße von 4 KB. Bitte begründen Sie Ihre Antwort.

(How many MB is the maximal partition size of a FAT16 file system with a block size of 4 KB. Please explain your result.)

256 MB

### Aufgabe 2 (12P) Reguläre Ausdrücke

Welche Ausgabe liefert folgendes Python-Skript?

(Please give the output of the following Python script.)

```
import re
zeichenk = '30955 pts/4 10:00:00 ps'
mu = ('5*5', '9*5+5', 'p*s+', 'p+?t??s?', '0.0?', '0[0:]*')
for item in mu:
    print item, ":", re.search(item, zeichenk).group()
```

5*5	:	55	✓
9*5+5	:	955	✓
p*s+	:	s	✓
p+?t??s?	:	pts	✓
0.0?	:	09	✓
0[0:]*	:	0	✓

30955 pts/4 10:00:00 ps

### Aufgabe 3: Ausnahmebehandlung (4 Punkte)

Welche Ausgabe liefert folgendes Python-Skript?  
(Please give the output of the following Python script.)

```
tests = (66,1), (66, 0.0), (69, 0), (25, 5)
try:
    try:
        for a, b in tests:
            try:
                print(a, b)
                print(a/b)
            finally:
                print("Division ausgeführt")
    except:
        print("Allgemeiner Fehler")
        raise ZeroDivisionError
except ZeroDivisionError:
    print("Nicht durch Null teilen ...")
finally:
    print("Aufräumen.")
print("Ende.")
```

### Aufgabe 4: Semaphore (8 Punkte)

Beschreiben Sie die beiden Hauptmethoden bei der Verwendung von Semaphoren.  
(Please describe the two main methods for using semaphores.)

### Aufgabe 5: Parameter Parsen (6 Punkte)

Schreiben Sie ein Shell-Skript, welches von den beim Aufruf übergebenen Zahlen ausgibt, die gerade oder kleiner 42 und mindestens 25 sind.

(Please write a shell script which accepts numbers as parameter and outputs the numbers which are even or smaller than 42 and at least 25.)

### Aufgabe 6: Shell-Programmierung (12 Punkte)

Schreiben Sie ein möglichst effektives Shell-Skript, welches den grössten gemeinsamen Teiler von zwei als Parameter übergebenen Zahlen ausgibt.

(Please write a shell script which outputs the greatest common divisor of two numbers which are given as parameter.)

*Bitte verwenden Sie für die Lösung der Aufgabe nur Befehle der Standard-Shell.  
(Please use only standard shell commands.)*

Bearbeitungszeit: 90 Minuten, Gesamtpunktzahl: 46, Viel Erfolg!

$$\begin{aligned} ① \quad & 2^{16} \cdot 4 \text{ KB} \\ & 2^6 \cdot 4 \text{ MB} \\ & 64 \cdot 4 \text{ MB} = \underline{\underline{256 \text{ MB}}} \end{aligned}$$

Man rechnet 2 hoch (FAT) 12, 16 oder 32, in dem Fall 16 und multipliziert das mit der Blockgröße.

② siehe Klausurblatt

④ Semaphore zeigen an, ob für einen Prozess noch ein Platz frei ist oder nicht. Die zwei Hauptmethoden sind Probieren und Freigeben (Probieren und unigeben).

Probieren:

WENN (Semaphore-Zähler = 0)

DANN schlafe

SONST Semaphore-Zähler -- und betrete kritischen Bereich

Freigeben:

WENN (schlafender Prozess vor Semaphore)

DANN wecke Prozess

SONST Semaphore-Zähler ++

Wenn kein Platz frei ist, dann schlafe, sonst ~~setze~~ ~~den~~ Zähler eins ab, und betrete den kritischen Bereich.

Wenn ein schlafender Prozess vor Semaphore ist, wecke diesen, sonst erhöhe den Zähler um eins.

③ ~~66~~ (66, 1)

66

Division ausgeführt ✓

66, 0.0

Division ausgeführt ✓

Allgemeiner Fehler

Nicht durch Null teilen...

Aufräumen.

Ende.

⑤

for para in ~~###~~ \$@

do

if [ \$(expr(\$para % 2) - eq 0) ]

then

echo ~~###~~ \$para

else

if [ \$para -ge 25 -a \$para -lt 42 ]

then

echo \$para ✓

fi

fi

done

⑥

x = ~~1~~ \$1

y = \$2

while [~~\$x~~ -ne ~~1~~ \$y]

do

if [~~\$x~~ -gt \$y]

then

x = (expr \$x - \$y)

else

~~y = expr \$y - \$x~~  
fi

done

echo ~~1~~ \$x

10

8  
-  
-  
-)