

## 1. Teilklausur (2011)

**Name:** ..... **Vorname:** ..... **Matrikel-Nummer:** .....

Ich bin mit der Veröffentlichung der Klausurergebnisse mit Matrikel-Nummer und Note im Internet einverstanden:

ja     nein

**Unterschrift:** .....

**Die maximale Punktzahl ist 52.**

<b>Aufgabe</b>	<b>A1</b>	<b>A2</b>	<b>A3</b>	<b>A4</b>	<b>A5</b>	<b>A6</b>	<b>A7</b>		<b>Summe</b>	<b>Note</b>
<b>Punkte</b>										

**Viel Erfolg!**

**1.Aufgabe** (4 Punkte)

Was ist der Wert und Typ folgender Python-Ausdrücke?

**Ausdruck**

**Wert**

**Typ**

`(7&0)==0`

`(2+3j)/2`

`[5 for i in range(3)]`

`{x for x in (('1', 1), ('0', 0))}`

## 2. Aufgabe (7 Punkte)

Programmieren Sie nur mit Hilfe von **if-else**-Anweisungen und logischen Ausdrücken den Inhalt der Funktion **decide\_color**, so dass folgendes Bild reproduziert wird.



Die **decide\_color** Funktion entscheidet, welche Farbe ein Pixel an einer beliebigen **x, y** Koordinate haben soll. Die **decide\_color** Funktion wird innerhalb einer **paint**-Funktion für jede **x, y** Koordinate aufgerufen, so dass ein quadratisches Bild mit Seitenlänge **size** gezeichnet wird. Nehmen Sie an, dass Sie nur folgende Farbnamen zur Verfügung haben:

`['black', 'gray', 'white']`

```
def decide_color (x, y, size):
```

### 3. Aufgabe (16 Punkte)

- a) Schreiben Sie unter Verwendung einer **while**-Schleife eine iterative Python-Funktion **dec2bin**, die bei Eingabe eines ganzzahligen Wertes **n** die Binärdarstellung der Zahl in eine Liste von Zeichen (String) zurückgibt.

Beispiel:

```
>>> dec2bin( 10 )
>>> 1010
```

- b) Schreiben Sie eine rekursive Version der **dec2bin**-Funktion, die das gleiche berechnet.

c) Analysieren Sie mit Hilfe der O-Notation die Komplexität Ihrer Funktionen.

d) Welches Hauptproblem gibt es in der *Python Virtuellen Maschine* mit endrekursiven Funktionen?

**4. Aufgabe** (9 Punkte)

Betrachten Sie die Zahlen innerhalb folgendes Arrays

$$A = [7, 3, 2, 0, 9, 4, 1, 6]$$

Die Zahl an der Position  $A[0]$  entspricht der Größe des Arrays ab der Position  $A[1]$ .

- a) Erläutern Sie kurz die wichtigsten Teile des Heapsort-Algorithmus.
- b) Schreiben Sie den Zustand des Arrays nach jeder Veränderung der Zahlenpositionen auf, wenn Sie die Zahlen ab der Position  $A[1]$  mit Hilfe des Heapsort-Algorithmus sortieren.
- c) Ist der Heapsort-Algorithmus stabil? Begründen Sie Ihre Antwort.

### 5. Aufgabe (1 Punkt)

Welches der Prädikate in der unten angegebenen **while**-Regel nennt man Invariante der **while**-Schleife?

$$\text{while-Regel: } \frac{\{P \wedge B\} S \{P\}}{\{P\} \text{ while } B \text{ do } S \{P \wedge \neg B\}}$$

Antwort: .....

### 6. Aufgabe (4 Punkte)

Gegeben sei folgendes Python-Programm, wobei K und M beliebige Integer-Werte mit  $K \geq 1$  und  $M \geq 1$  sind, die eingegeben werden und als Konstanten betrachtet werden:

```
K = int(input("K = "))
M = int(input("M = "))
```

```
p = K
e = 0
```

```
while p <= M:
    p = p * K
    e = e + 1
```

Welche der folgenden Prädikate stellen eine Invariante der **while**-Schleife dar?

- $p = K^{e+1} \wedge K^{e-1} \leq M \wedge K > 0$
- $p = K^e \wedge K^{e-1} \leq M \wedge K > 1$
- $p = K^{e+1} \wedge K^e < M \wedge K > 0$
- $p = K^{e+1} \wedge K^e \leq M \wedge K > 0$
- $p = K^{e+1} \wedge K^e \leq M \wedge K > 1$
- $p = K^e \wedge K^{e-1} < M \wedge K > 0$

Welche der Invarianten garantiert die Terminierbarkeit? Begründen Sie Ihre Antwort.

### 7. Aufgabe (7 Punkte)

Beweisen Sie die Gültigkeit folgender Programmformel:

```
# { z > 0 }  
if x < 0:  
    x = z - x  
else:  
    x = x + 1  
# { z > 0 ∧ x > 0 }
```