

## Algorithmen und Programmierung Klausur zum WS 2003/2004

1. Aufgabe:

2 + 2 + 1 = 5 Punkte

- Geben Sie die periodische Dualzahl  $0, \overline{001}$  als Bruch im Dezimalsystem an.
- Transformieren Sie die Zahl  $(0,35)_{10}$  in eine Dualzahl.
- Betrachten Sie die Zweier-Komplementdarstellung auf  $n$  Bits und ergänzen Sie den Satz auf dem Lösungsblatt.

2. Aufgabe:

2 + 7 = 9 Punkte

Gegeben sei folgendes Programmstück:

```
// Eingabe  $x > 0$ , ganzzahlig  
// Eingabe  $y \geq 0$ , ganzzahlig  
z = 0;  
  
while ( y >= x ) {  
    y = y - x;  
    z = z + 1;  
}  
  
// Ergebnis steht in z
```

- Was berechnet dieses Programmstück (Wert der Variablen  $z$  nach Abarbeitung)?
- Weisen Sie die Korrektheit Ihrer Aussage aus a) nach. Gehen Sie dabei gemäß dem Muster
  - Aussage über die Endlichkeit der Anzahl der Schleifendurchläufe (1)
  - Beweis einer geeigneten Invariante (Sie dürfen bei Bedarf die Operatoren  $\text{DIV}(x, y)$  (ganzzahlige Division von  $x$  durch  $y$ ) und  $\text{MOD}(x, y)$  (Rest von  $\text{DIV}(x, y)$ ) benutzen.) (4)
  - Quintessenz (2)

vor.

3. Aufgabe:

1 + 7 = 8 Punkte

Gegeben sei folgendes Programmstück:

```
// Eingabe  $y \geq 1$ , ganzzahlig
x = 1;

while (y > 0) {
    x = x * 3;
    y = y - 1;
}

// Ergebnis steht in x
```

- Was berechnet dieses Programmstück (Wert der Variablen  $x$  nach Abarbeitung)?
- Weisen Sie die Korrektheit Ihrer Aussage aus a) nach. Gehen Sie dabei gemäß dem Muster
  - Aussage über die Endlichkeit der Anzahl der Schleifendurchläufe (1)
  - Beweis einer geeigneten Invariante (4)
  - Quintessenz (2)

vor.

4. Aufgabe:

2 + 7 + 2 = 11 Punkte

Betrachten Sie das Problem der „Türme von Hanoi“ mit 3 Stäben ( $A, B, C$ ). Die Scheiben seien mit 1 (oberste, kleinste Scheibe) bis  $n$  (unterste, größte Scheibe) bezeichnet. Sie sollen von  $A$  nach  $B$  umgesetzt werden.

Zusätzlich muss folgende Einschränkung eingehalten werden: Das Umsetzen von Scheiben direkt von  $A$  nach  $B$  und umgekehrt von  $B$  nach  $A$  ist **nicht** erlaubt!

- Zeichnen Sie für die vorgegebene Situation auf dem Lösungsblatt jeweils die Zwischen-Situationen nach jedem einzelnen Schritt bis zum Ende (alles auf Stab  $B$ ) auf.
- Tragen Sie in den Rahmen auf dem Lösungsblatt eine rekursive Methode mit **genau drei** rekursiven Aufrufen zur Ermittlung einer zulässigen Schrittfolge zur Umsetzung von  $n$  Scheiben ein. Für einen Schritt soll das Programm einen Text der Art „Scheibe  $i$  von ... nach ...“ ausgeben. Hinweis: Als Hilfsstab müssen Sie immer  $C$  benutzen.
- Geben Sie eine nicht-rekursive Formel für die Anzahl der Umsetzungen (Züge) Ihrer Methode aus b) in Abhängigkeit von  $n$  an.

## 5. Aufgabe:

4 Punkte

Gegeben sei ein aufsteigend sortiertes Feld  $f$  ganzer Zahlen. Betrachten Sie folgende Version der binären Suche nach einem Element  $x$ .

```

int a , b , h;

a = 0;
b = f.length - 1;

while ( b > a ) {
    h = ( a + b ) / 2;
    if ( x <= f[h] )
        b = h;
    if ( x >= f[h] )
        a = h;
}

if ( f[a] == x )
    // Ausgabe : x an Position a enthalten
else
    // Ausgabe : x nicht enthalten

```

Warum ist der Algorithmus nicht für alle möglichen Eingaben von  $x$  korrekt? Begründen Sie Ihre Aussage durch ein Beispiel!

## 6. Aufgabe:

5 + 3 = 8 Punkte

a) Finden Sie alle Fehler in folgendem Programmstück:

```

int i;
boolean [] feld;
short x;

for ( i = 0; i <= feld.length; i++)
    feld[i] = 1;

x = x + 1;

```

b) Was gibt folgendes Programm aus?

```
public class Test{  
  
    static int x=1 , y=2 , z=3;  
  
    public static void up(int x){  
        int y=5;  
        System.out.println(x);  
        System.out.println(y);  
        System.out.println(z);  
        z=4;  
    }  
  
    public static void main(String [] args){  
        int y=10;  
        up(y);  
        System.out.println(x);  
        System.out.println(y);  
        System.out.println(z);  
    }  
}
```