

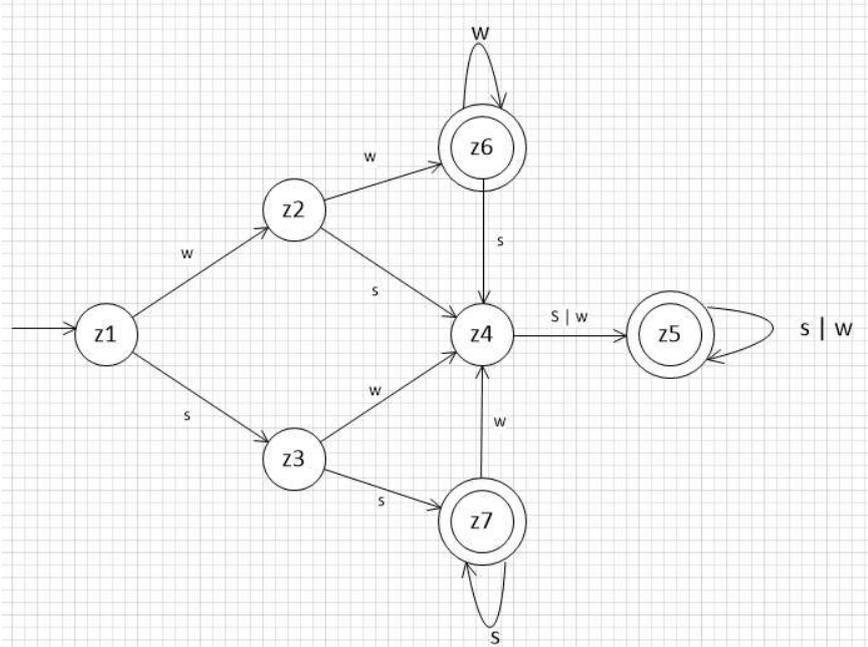
# Informatik Abitur Bayern 2016 / IV - Beispiellösung

Autor:  
Ober

1a Bei i) handelt es sich um einen Crystalball-Stab, da er aus mehr als zwei Kugeln besteht und die oberste Farbe (schwarz) bereits weiter unten (an zweiter Stelle) bereits vorkommt. 2

Bei ii) handelt es sich nicht um einen Crystalball-Stab, da die oberste Farbe (schwarz) nicht weiter unten bereits vorkommt.

1b



8

1c  $\langle \text{Stab} \rangle \rightarrow s \langle S \rangle \rightarrow sw \langle Z \rangle \rightarrow sws$  ist die einzige Möglichkeit, ein Wort mit der Zeichenfolge sws am Beginn abzuleiten. 3

1d

$\langle \text{Stab} \rangle \rightarrow s \langle S \rangle \mid w \langle W \rangle$   
 $\langle S \rangle \rightarrow s \langle S \rangle \mid w \langle Z \rangle \mid s$   
 $\langle W \rangle \rightarrow w \langle W \rangle \mid s \langle Z \rangle \mid w$   
 $\langle Z \rangle \rightarrow s \mid w \mid s \langle Z \rangle \mid w \langle Z \rangle$

3

1e Unter Verklemmung versteht man einen Zustand, bei dem eine zyklische Wartesituation zwischen mehreren Prozessen auftritt, wobei jeder Prozess auf die Freigabe von Betriebsmitteln wartet, die von einem anderen Prozess bereits exklusiv belegt sind. Bei einer Pflanzaktion könnte es wie folgt zu einer Verklemmung kommen:  
In einem Schrank liegen ein Beutel Samen und eine Schaufel. Zwei Gärtner wollen die Samen einpflanzen. Der eine Gärtner nimmt den Beutel, der andere die Schaufel. Der Gärtner mit den Samen kann nicht anpflanzen, da er ohne die Schaufel keine Löcher graben kann. Der andere Gärtner kann zwar Löcher graben, aber nichts anpflanzen, weil er keine Samen hat. 4

2a

```

public int anzahlGeben(int n)
{
    int x;
    if (n==0)
    {
        x=1;
    }
    else
    {
        x = 2*anzahlGeben(n-1);
    }
    return x;
}

```

3

2b

$$\begin{aligned}
 \text{anzahlGeben}(3) &= 2 * \text{anzahlGeben}(2) \\
 &= 2 * 2 * \text{anzahlGeben}(1) \\
 &= 2 * 2 * 2 * \text{anzahlGeben}(0) \\
 &= 2 * 2 * 2 * 1 = 8
 \end{aligned}$$

2

Anzahl der Durchläufe: 4

2c

n	Anzahl der Durchläufe
0	1
1	2
2	3
3	4
4	5

3

Laufzeit:

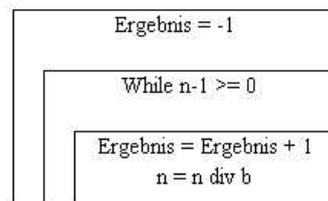
Da die Anzahl der Durchläufe immer um 1 größer ist als die Inkubationszeit n, ergibt sich ein lineares Laufzeitverhalten ( $O(n)$ ).

3a Speicherzelle 100: Ausgabe  
 Speicherzelle 101: Zahl n  
 Speicherzelle 102: Basis b

5

100	101	102
	25	3
-1	25	3
0	25	3
0	8	3
1	8	3
1	2	3
2	2	3
2	0	3

3b



5

3c 5 (Befehl aus dem Speicher holen)

2

- 2 (Befehl decodieren)
- 6 (Operandenadresse bestimmen)
- 1 (Operanden holen)
- 7 (Befehl ausführen)
- 4 (Ergebnis im Akkumulator ablegen)
- 3 (Befehlszähler erhöhen)