

Informatik Abitur Bayern 2011 / IV - Beispiellösung

Autor:
Weiler

1a Akzeptiert werden 2500, -13 und +3. Nicht akzeptiert wird 0 (hier verbleibt der Automat im Startzustand). 2

1b Der Automat erkennt folgende Sprache: 4

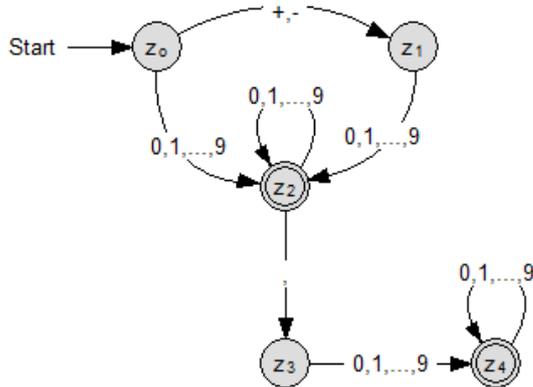
$\langle \text{Zahl} \rangle ::= [\langle \text{Vorzeichen} \rangle] \langle \text{ZifferUngleichNull} \rangle \{ \langle \text{Ziffer} \rangle \}$

$\langle \text{Vorzeichen} \rangle ::= +|-$

$\langle \text{ZifferUngleichNull} \rangle ::= 1|2|3|4|5|6|7|8|9$

$\langle \text{Ziffer} \rangle ::= 0|\langle \text{ZifferUngleichNull} \rangle$

1c 5



2a Nach Ablauf des Programms befinden sich folgende Werte in den Speicherzellen 98 bis 100: 5

98	99	100
18	0	6

(Nach der ersten Abarbeitung der Zeilen 1 – 9 befinden sich folgende Werte in den Speicherzellen 98 bis 100:

98	99	100
6	2	6

Nach der zweiten Abarbeitung der Zeilen 3 – 9 befinden sich folgende Werte in den Speicherzellen 98 bis 100:

98	99	100
12	1	6

)

2b Alternative 1:



Alternative 2:

4



```

2c 1   loadi 1   //Startwert für die Potenz
    2   store 97
    3   load 97  //In Zelle 97 befindet sich das Zwischenergebnis
    4   store 99 //Das Zwischenergebnis wird nach Zelle 99 übertragen
    5   loadi 0  //Multiplikation von Zellwert 99 und 100 → 2a) (Beginn)
        //→ Multiplikation von Zwischenergebnis und der Zahl a
    6   store 98
    7   load 98
    8   add 100
    9   store 98
   10  load 99
   11  dec
   12  store 99
   13  jg 7      //Multiplikation von Zellwert 99 und 100 → 2a) (Ende)
   14  load 98  //neues Zwischenergebnis (Produkt aus Zellwert 99 und
        //100 befindet sich in Zelle 98 → vgl. Aufgabe 2a))
   15  store 97 //neues Zwischenergebnis wird nach Zelle 97 übertragen
   16  load 96
   17  dec      //Der Exponent wird um 1 verringert
   18  store 96
   19  jg 3     //Falls der Exponent noch größer 0 ist, wird die
        //Multiplikation (neues Zwischenergebnis mit der Zahl a)
        //weiter fortgesetzt
   20  end

```

13

Die Berechnung der Potenz a^n wird durch fortgesetzte Multiplikation realisiert. Dabei wird die Zahl a bzw. eine Potenz der Zahl a: a^2, a^3, \dots, a^{n-1} n-Mal mit der Zahl a multipliziert. Die Multiplikation des jeweiligen Zwischenergebnisses mit der Zahl a wird analog zu dem Programm aus Aufgabe 2a) realisiert (vgl. Speicherzellen 5 -13).

Speicherzellenbelegung:

Die Zahl a befindet sich in Zelle 100.

Der Exponent n befindet sich zu Beginn in Zelle 96. Im Laufe des Programms wird der Exponent (Zähler) bei jedem Durchlauf um eins verringert. Nimmt der Zähler den Wert 0 an, so stoppt das Programm und der Potenzwert befindet sich in der Zelle 97.

3a Unter einer **Verklemmung** versteht man in der Informatik einen Zustand, bei dem eine 5
zyklische Wartesituation zwischen mehreren Prozessen auftritt, wobei jeder beteiligte Prozess auf die Freigabe von Betriebsmitteln wartet, die ein anderer beteiligter Prozess bereits exklusiv belegt hat.

Wenn die Fahrzeuge gleichzeitig in die Kreuzung befahren (exklusives Betriebsmittel), so blockieren sie sich gegenseitig. Jedes Fahrzeug (Prozess) wartet aufgrund der konkurrierenden Verkehrsregeln auf die Freigabe der Kreuzung, die von einem anderen Fahrzeug gleichzeitig „belegt“ wird. Fahrzeug A wartet darauf, dass Fahrzeug C die Kreuzung überquert und somit freigibt. Fahrzeug B wartet darauf, dass Fahrzeug A abbiegt. Fahrzeug C wartet darauf, dass Fahrzeug B die Kreuzung überquert. Durch das gegenseitige Blockieren entsteht somit eine Verklemmung.

3b Mit **Monitoren** können Verklemmungen verhindert werden. 2
Mit Monitoren lassen sich geschützte Programmabschnitte definieren, die immer nur von einem Prozess gleichzeitig ausgeführt werden dürfen. In diesem Beispiel entspricht der zu schützende Bereich der Methode KreuzungBefahren(), die jeweils nur von einem Fahrzeug genutzt werden darf. (Alternativ kann auch mit dem Konzept der **Semaphoren** gearbeitet werden. Mit Hilfe einer binären Semaphore muss der exklusive Zugriff auf die Kreuzung sichergestellt werden.)