

Informatik A, Sommersemester 2005

Gudrun Fischer, André Schaefer
 infoa@is.informatik.uni-duisburg.de

Musterlösung Übungsblatt 8

Aufgabe 24: Von Mini-Pascal zur Stapelmaschine

- (a) Übersetzt das angegebene Mini-Pascal-Programm in Stapelmaschinencode.

Adresse	Befehl	Kommentar
		PComp(Aufgabe24)
00	JMP 27	Springe zum Anfang des Hauptprogrammes
		BComp(Procedur q)
01	LOAD 1,1	Lade t, gehe dabei ein Level zurück
02	STORE 0,0	Zuweisung zum lokalen x
03	CONST 0	EComp(0)
04	STORE 0,2	Zuweisung von z
05	LOAD 0,0	
06	CONST 0	EComp(0)
07	SUB	
08	JLE 24	Anfang der while-Schleife, abbruch falls $x \leq 0$
09	LOAD 0,0	lege x auf Stapel
10	CONST 10	EComp(10)
11	DIV	teile
12	STORE 0,1	weise das Ergebnis y zu
13	LOAD 0,2	lege z auf Stapel
14	LOAD 0,0	lege x auf Stapel
15	ADD	Addition $z+x$
16	CONST 10	EComp(10)
17	LOAD 0,1	lege y auf Stapel
18	MUL	Multiplikation $(10*y)$
19	SUB	Subtraktion $(z+x)-(10*y)$
20	STORE 0,2	Zuweisung $z:=(z+x)-(10*y)$
21	LOAD 0,1	lege y auf Stapel
22	STORE 0,0	weise x den Wert zu
23	JMP 05	Springe zum Schleifenanfang
24	LOAD 0,2	Lade z
25	STORE 1,2	Weise den Wert dem globalen r zu
26	RET	Ende von q

		BComp(Hauptprogramm)
27	CONST 25	EComp(25)
28	STORE 0,0	Zuweisung zu x
29	LOAD 0,0	
30	STORE 0,1	Zuweisung von t
31	CALL 0,01,3	Aufruf von q
32	RET	Ende des Hauptprogrammes

7 Punkte

- (b) Gebt die einzelnen Zustände der Stapelmaschine bei der Ausführung des entstandenen Codes an (also für jede ausgeführte Zeile SM-Code jeweils die Inhalte des Instruktionszählers, des Datenstapels und des Prozedurstapels).

i	z_i	Befehl	δ_{i+1}	π_{i+1}	z_{i+1}	Kommentar
01	00	JMP 27		$\langle 0,0,(0,0,0) \rangle$	27	Springe zum Anfang des Hauptprogrammes
02	27	CONST 25	25	$\langle 0,0,(0,0,0) \rangle$	28	Zuweisung der Variablen
03	28	STORE 0,0		$\langle 0,0,(25,0,0) \rangle$	29	
04	29	LOAD 0,0	25	$\langle 0,0,(25,0,0) \rangle$	30	
05	30	STORE 0,1		$\langle 0,0,(25,25,0) \rangle$	31	
06	31	CALL 0,01,3		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,0) \rangle$	01	Aufruf der Prozedur q
07	01	LOAD 1,0	25	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,0) \rangle$	02	
08	02	STORE 0,0		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	03	Zuweisung zum lokalen x
09	03	CONST 0	0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	04	
10	04	STORE 0,2		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	05	
11	05	LOAD 0,0	25	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	06	Erzeuge Bedingung für While-Schleife
12	06	CONST 0	25.0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	07	
13	07	SUB	25	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	08	
14	08	JLE 24		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	09	Prüfe Bedingung der While-Schleife
15	09	LOAD 0,0	25	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	10	
16	10	CONST 10	25.10	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	11	
17	11	DIV	2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,0,0) \rangle$	12	
18	12	STORE 0,1		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	13	
19	13	LOAD 0,2	0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	14	$z:=z+x-10*y$
20	14	LOAD 0,0	0.25	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	15	
21	15	ADD	25	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	16	
22	16	CONST 10	25.10	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	17	
23	17	LOAD 0,1	25.10.2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	18	
24	18	MUL	25.20	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	19	
25	19	SUB	5	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,0) \rangle$	20	
26	20	STORE 0,2		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,5) \rangle$	21	
27	21	LOAD 0,1	2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(25,2,5) \rangle$	22	
28	22	STORE 0,0		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	23	
29	23	JMP 5		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	05	Springe zum Anfang der While-Schleife

i	z_i	Befehl	δ_{i+1}	π_{i+1}	z_{i+1}	Kommentar
30	05	LOAD 0,0	2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	06	Erzeuge Bedingung für While-Schleife
31	06	CONST 0	2.0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	07	
32	07	SUB	2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	08	Prüfe Bedingung der While-Schleife
33	08	JLE 24		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	09	
34	09	LOAD 0,0	2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	10	z:=z+x-10*y
35	10	CONST 10	2.10	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	11	
36	11	DIV	0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,2,5) \rangle$	12	
37	12	STORE 0,1		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	13	
38	13	LOAD 0,2	5	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	14	
39	14	LOAD 0,0	5.2	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	15	
40	15	ADD	7	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	16	
41	16	CONST 10	7.10	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	17	
42	17	LOAD 0,1	7.10.0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	18	
43	18	MUL	7.0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	19	
44	19	SUB	7	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,5) \rangle$	20	
45	20	STORE 0,2		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,7) \rangle$	21	
46	21	LOAD 0,1	0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(2,0,7) \rangle$	22	
47	22	STORE 0,0		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	23	
48	23	JMP 5		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	05	Springe zum Anfang der While-Schleife
49	05	LOAD 0,0	0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	06	Erzeuge Bedingung für While-Schleife
50	06	CONST 0	0.0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	07	x=0, springe nach 31
51	07	SUB	0	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	08	
52	08	JLE 24		$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	24	
53	24	LOAD 0,2	7	$\langle 0,0,(25,25,0) \rangle . \langle 32,1,(0,0,7) \rangle$	25	
54	25	STORE 1,2		$\langle 0,0,(25,25,7) \rangle . \langle 32,1,(0,0,7) \rangle$	26	
55	26	RET		$\langle 0,0,(25,25,7) \rangle$	32	
56	32	RET		(leer)		Ende des Hauptprogrammes, Rückkehr zum System

6 Punkte

- (c) Was wird in der Prozedur q berechnet und steht nach der Ausführung von q in r? 2 Punkt
 Die Quersumme der Zahl, die dem globalen x am Anfang zugewiesen wird.
 In diesem Fall ist es die Zahl 7.

Zu beachten ist hier bei den Befehlen Load und Store folgendes:

- Die erste Zahl gibt den Ort der Variable (Level) an. Ist sie 0, so wird die aktuelle (lokale) Variablenmenge referenziert, ist sie n mit $n > 0$, so referenziert der Befehl die n Level höher liegende Menge (also bei n=1 die Variablen der aufrufenden Schicht usw.).
- Die zweite Zahl gibt die Position der Variablen in der Variablenmenge, beginnend mit 0 an.