

Professor Dr.-Ing. Stefan Kowalewski
Dipl.-Inform. Dominik Franke
Dipl.-Inform. Sebastian Biallas

Aachen, 17. Dezember 2010
SWS: V4/Ü2, ECTS: 7

Einführung in die Technische Informatik

WS 2010/2011

Blatt 9: Musterlösung

ACHTUNG: Die Musterlösung ist ein zusätzliches Serviceangebot. Sie erhebt weder Anspruch auf Vollständigkeit noch auf Korrektheit.

Aufgabe 1: (★) Arithmetik

Aufgabe 2: (★) Bitweise Operationen

Aufgabe 3: (★) Speicherzugriffe

Aufgabe 4: (★) Absolutbetrag

Aufgabe 5: (★) Hammingabstand

Aufgabe 6: Programmieren

Schreiben Sie ein MMIX-Programm, das einen Rechner nach umgekehrter polnischer Notation implementiert.

Lösungsvorschlag

	LOC	Data_Segment		BP	temp,2F
stackPtr	GREG	0		SUB	temp,chr,48
x0	GREG	@		SET	haveValue,1
X0	IS	@		MUL	value,value,10
N	IS	1024		ADD	value,value,temp
				JMP	1B
	LOC	#100			
			2H	BZ	haveValue,3F
charEnd	IS	0		* SET	\$6,value
				PUSHJ	4,pushValue
charNL	IS	10		SET	haveValue,0
charPlus	IS	43		SET	value,0
charMinus	IS	45			
charStar	IS	42	3H	CMP	temp,chr,charSpace
charSlash	IS	47		BZ	temp,1B
charZero	IS	48			
charNine	IS	57		CMP	temp,chr,charPlus
charSpace	IS	32		BNZ	temp,3F
				PUSHJ	5,add
ptr	IS	\$0		JMP	4F
chr	IS	\$1			
size	IS	\$2	3H	CMP	temp,chr,charMinus
temp	IS	\$3		BNZ	temp,3F
haveValue	IS	\$4		PUSHJ	5,subtract
value	IS	\$5		JMP	4F
Main	SET	value,0	3H	CMP	temp,chr,charStar
	SET	stackPtr,1024		BNZ	temp,3F
				PUSHJ	5,multiply
	GETA	\$255,9F		JMP	4F
	TRAP	0,Fgets,StdIn			
	BN	\$255,Error	3H	CMP	temp,chr,charSlash
	BZ	\$255,Exit		BNZ	temp,3F
	SET	size,\$255		PUSHJ	5,divide
				JMP	4F
	SET	ptr,0			
	SET	haveValue,0	3H	JMP	Error
1H	LDBU	chr,x0,ptr	4H	CMPU	temp,ptr,size
	ADD	ptr,ptr,1		BN	temp,1B
	CMPU	temp,chr,charEnd		JMP	Main
	BZ	temp,Main			
	CMPU	temp,chr,charNL	Done	BZ	haveValue,3F
	BZ	temp,Done		PUSHJ	4,pushValue
	CMPU	temp,chr,charZero	3H	PUSHJ	6,popValue
	BN	temp,2F			
	CMPU	temp,chr,charNine		SET	value,\$6

	BNN	value,1F			
	GETA	\$255,Minus	subtract	GET	\$0,rJ
	TRAP	0,Fputs,StdOut		PUSHJ	1,popValue
	NEG	value,value		PUSHJ	2,popValue
1H	SWYM			SUB	\$2,\$2,\$1
				PUSHJ	1,pushValue
	SET	ptr,10		PUT	rJ,\$0
	SET	temp,0		POP	0,0
	STBU	temp,x0,ptr			
2H	SUB	ptr,ptr,1	multiply	GET	\$0,rJ
	DIV	value,value,10		PUSHJ	1,popValue
	GET	temp,rR		PUSHJ	2,popValue
	ADD	temp,temp,charZero		MUL	\$2,\$2,\$1
	STBU	temp,x0,ptr		PUSHJ	1,pushValue
	BNZ	value,2B		PUT	rJ,\$0
	LDA	\$255,x0,ptr		POP	0,0
	TRAP	0,Fputs,StdOut			
			divide	GET	\$0,rJ
	GETA	\$255,NewLine		PUSHJ	1,popValue
	TRAP	0,Fputs,StdOut		PUSHJ	2,popValue
				DIV	\$2,\$2,\$1
Exit	SET	\$255,0		PUSHJ	1,pushValue
	TRAP	0,Halt,0		PUT	rJ,\$0
Error	SET	\$255,0		POP	0,0
	SUB	\$255,\$255,1			
	TRAP	0,Halt,0			
9H	OCTA	X0,N			
Minus	BYTE	"-",0,0,0			
NewLine	BYTE	10,0,0,0			
pushValue	STOU	\$0,x0,stackPtr			
	ADD	stackPtr,stackPtr,8			
	POP	0,0			
popValue	SUB	stackPtr,stackPtr,8			
	LDOU	\$0,x0,stackPtr			
	POP	1,0			
add	GET	\$0,rJ			
	PUSHJ	1,popValue			
	PUSHJ	2,popValue			
	ADD	\$2,\$2,\$1			
	PUSHJ	1,pushValue			
	PUT	rJ,\$0			
	POP	0,0			