

Übungszettel 1

Aufgabe 1: Maschinenprogramme

Der Ausdruck

```
x=5;  
if (x < y)  
    x=y;  
else  
    y=x;
```

soll mit Hilfe der folgenden Assemblerbefehle umgesetzt werden.

LOAD x Lade den Inhalt von x in den Akkumulator
STORE x Speichere den Inhalt des Akkumulators nach x
ADD x Addiere x zum Inhalt des Akkumulators
SUB x Subtrahiere x vom Inhalt des Akkumulators
JMPNEG x Springe zur Marke x , wenn der Inhalt des Akkumulators < 0

Zu dem Code soll auch die entsprechende Speicherbelegung angegeben werden. Dabei wird im Assembler x als Marke für die Speicherzelle mit dem Inhalt von x verwendet, während im Maschinencode die tatsächliche Adresse eingetragen wird.

Aufgabe 2: Mikro-Instruktionen

Der Maschinenbefehl *JMPNEG* soll als Folge von CPU-Mikro-Instruktionen kodiert werden. Dessen Bedeutung ist folgendermaßen:

JMPNEG Adr Springe zur Adresse Adr , falls der Inhalt des Akkumulators < 0

Es kann davon ausgegangen werden, dass der Operator in der Zelle n und der Operand in der Zelle $n+1$ stehen:

n:	<table border="1"><tr><td><i>JMPNEG</i></td></tr></table>	<i>JMPNEG</i>
<i>JMPNEG</i>		
n+1:	<table border="1"><tr><td><i>Adr</i></td></tr></table>	<i>Adr</i>
<i>Adr</i>		

Anmerkung: Wenn notwendig, kann auch Pseudocode wie *dekodiere Befehl* verwendet werden!