Maschinenorientierte Programmierung

PC Inside 2021 Aufgaben Name:_____

1. Die Inhalte der zehn 1-Byte-Speicherplätze ab Adresse 100H sollen in die zehn 1-Byte-Speicherplätze ab 125H kopiert werden.

2. Es soll die Summe aller Byte-Inhalte der Speicherplätze 18H bis 28H gebildet und das Ergebnis in 30H abgelegt werden.

3. Eine Zahl zwischen 0 und 8 sei im Speicherplatz 7H (und 8H) als Datenwort gegeben. Berechnen Sie durch ein geeignetes Maschinenprogramm die Fakultät dieses Wertes, und speichern Sie das Ergebnis in 20H und 21H ab. (Nur mnemonische Darstellung!) Es gilt bekanntlich 0! = 1 und n! = 1 · 2 · ... · n .

4. a) Das folgende Maschinenprogramm ist von Zeile 4 bis 8 zu kodieren.
   b) Nach Ausführung des Befehls in Zeile 9 liegen folgende Registerinhalte vor:
   AX 00 E6
   BX 00 20
   CX FFFF
   DX 80 32
   Geben Sie an, wie das Overflow-Flag und das Carry-Flag bei der Abarbeitung der Programmzeilen 10 bis 13 gesetzt werden. Diese Angaben sind anschließend zu begründen.

O-Flag	C-Flag	Hexadezimale Kodierung	Mnemonik
			1 MOV BX,20H
			2 MOV SI,4
			3 MOV DI,-5
			4 SUB CX,CX
			5 MOV AL,[BX][DI][3]
			6 INC SI
			7 MOV DH,[20H][SI]
			8 MOV DL,[26H]
			9 DEC CX
			10 SUB CL,DH
			11 ADD DL,CL
			12 ADD DL,AL
			13 ADD AL,DL

Addition und Subtraktion:

3 Zyklen

```
ADD reg1,reg2

0000001w  11...... ........
```

add register reg2 to reg1
(reg1:= reg1 + reg2)

	w:	1	oder	0
000	für	AX	oder	AL
001	für	CX	oder	CL
010	für	DX	oder	DL
011	für	BX	oder	BL
100	für	SP	oder	AH
101	für	BP	oder	CH
110	für	SI	oder	DH
111	für	DI	oder	BH

3 Zyklen

```
SUB reg1,reg2

0010101w  11...... ........
```

sub register reg2 from reg1
(reg1:= reg1 - reg2)

Befehlsobjektcodes des MOV-Befehls bei verschiedenen Adressierungen:

Beispiel 1:

```
MOV reg1,data16/8
                 low      high
1011w...  ........(........)
```

move data immediate into register

kopiere umittelbare Daten in ein Register

Allgemein:

```
Opcode|W|REG  ........(........)
```

Beispiel 1:

```
MOV reg1,reg2

1000101w  11......(........)
```

move data from 2nd to 1st register

kopiere Daten vom 2. ins 1. Register

Allgemein:

```
Opcode|D|W| MOD|REG|R/M (........(........))
```

Abkürzungen:

D	destination	Ziel
W	word or byte	Wort oder Byte
MOD	mode	Modus
REG	register	Register
R/M	register or memory	Register oder Speicher

MOD-Feld und R/M-Feld:

MOD:	**00**	**01**	**10**	**11**
R/M				**w: 1 oder 0**
000	BX + SI	BX + SI + data8	BX + SI + data16	000 für AX oder AL
001	BX + DI	BX + DI + data8	BX + DI + data16	001 für CX oder CL
010	BP + SI	BP + SI + data8	BP + SI + data16	010 für DX oder DL
011	BP + DI	BP + DI + data8	BP + DI + data16	011 für BX oder BL
100	SI	SI + data8	SI + data16	100 für SP oder AH
101	DI	DI + data8	DI + data16	101 für BP oder CH
110	data16	BP + data8	BP + data16	110 für SI oder DH
111	BX	BX + data8	BX + data16	111 für DI oder BH

Erhöhen und Vermindern von 16/8-Bit-Registern:

2 Zyklen

```
INC reg16
01000...
```

increment register16

```
INC reg
1111111w 11000...
```

increment register8

w:	1	oder	0
000	für	AX	oder AL
001	für	CX	oder CL
010	für	DX	oder DL
011	für	BX	oder BL
100	für	SP	oder AH
101	für	BP	oder CH
110	für	SI	oder DH
111	für	DI	oder BH

2 Zyklen

```
DEC reg16
01001...
```

decrement register16

```
DEC reg
1111111w 11001...
```

decrement register8