

# Einführung in die Rechnerarchitektur

## Flags und bedingte Sprünge

**Lukas Hertel**

Lehrstuhl für Rechnerarchitektur und Parallele Systeme  
Fakultät für Informatik  
Technische Universität München

22. November 2021



*TUM Uhrenturm*

# Hausaufgabe



<b>Kurz</b>	<b>Name</b>	<b>Beschreibung</b>	<b>Bei CMP EAX, EBX 1 falls</b>
CF	Carry Flag	Übertrag	EAX < EBX (ohne Vorzeichen)
ZF	Zero Flag	Ergebnis ist null	EAX = EBX (Vorzeichen egal)
SF	Sign Flag	Vorzeichen	Register negativ
OF	Overflow Flag	Überlauf	EAX - EBX nicht repräsentierbar (mit Vorzeichen)

## Aufgabe 1a

	Befehlsfolge	Carry	Overflow	Sign	Zero
1:	-	0	1	0	1
2:	MOV EAX, 300	—	—	—	—
3:	SUB AL, 100	—	—	—	—
4:	MOV AX, 300	—	—	—	—
5:	CMP AH, 200	—	—	—	—
6:	ADD AL, 100	—	—	—	—

## Aufgabe 1b

	Befehlsfolge	Carry	Overflow	Sign	Zero
1:	-	0	0	0	1
2:	MOV EAX, -120	—	—	—	—
3:	PUSH AX	—	—	—	—
4:	CMP AL, 0	—	—	—	—
5:	ADD AL, 140	—	—	—	—

# Bedingte Sprünge

## Möglichkeiten um zur marke1 zu springen

## Von Java zu Assembler

Wo setzen wir die Marken?

```
int ebx;  
ebx = 50;  
while (ebx <= 60) {  
    fkt(ebx);  
    ebx = ebx + 1;  
}
```

## Multiplikation mit Shifts

- SHL/SAL/SHR  $n$ 
  - Shift um  $n$  Stellen
  - Wird mit 0en aufgefüllt
- SAR  $n$ 
  - Shift um  $n$  Stellen
  - Wird mit 1en aufgefüllt (Wieso?)
- Was passiert bei einem Shift um  $n$  Stellen?

# Verschiebung von Bits



# Einfügen von Bits in ein anderes Register

# Bits prüfen auf Bedingung

# Schnelle Division ohne div





# Nullsetzen von bit $n$

