

# Theorie 3

## Gruppe 35

Matias Heredia Novillo, Anastasia  
Paschalidou, Niclas Gregor

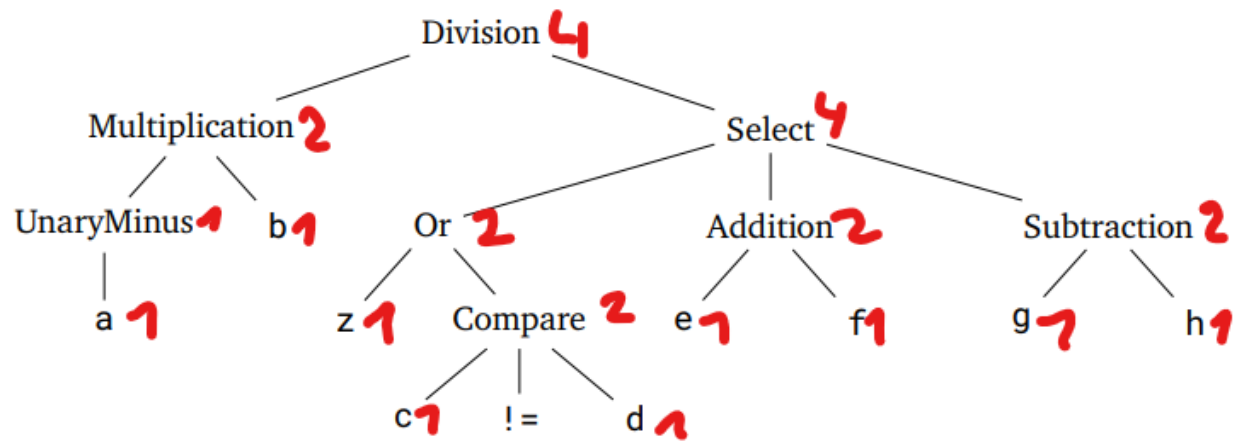
## Aufgab 3.1

a)

- Ausdruck =  $-a*b / (z \text{ or } c \neq d ? e+f : g-h)$
- Stack:

```
LOAD a
NEG
LOAD b
MUL
LOAD z
LOAD c
LOAD d
CPM_NE
OR
LOAD e
LOAD f
ADD
LOAD g
LOAD h
SUB
SEL
DIV
```

b)



c)

```
r1 = ld c
r2 = ld d
r1 = cpm_ne r1, r2
r2 = ld z
r1 = or r1, r2
r2 = ld e
r3 = 1
r2 = add r2, r3
r3 = ld g
r4 = ld h
r3 = sub r3, r4
r1 = sel r1, r2, r3
r2 = ld a
r2 = neg r2
r3 = ld b
r2 = mul r2, r3
r1 = div r2, r1
```

## Aufgab 3.2

a)

Vor Aufruf (19)

0[LB] / 0[SB]	0[SB]	Dynamic Link
1[LB] / 1[SB]	1[CB]	Return Address
2[LB] / 2[SB]	3	Speicherort von Ausdruck x
3[LB] / 3[SB]	7	Speicherort von Ausdruck y
4[LB] / 4[SB]	29	Speicherort von Ausdruck z

b)

Stackelement	Stack Inhalt	Bedeutung
-13[LB] / 0[SB]	0[SB]	Dynamic Link
-12[LB] / 1[SB]	1[CB]	Return Adresse von main
-11[LB] / 2[SB]	0	Speicherort von Variable x [0][0]
-10[LB] / 3[SB]	0	Speicherort von Variable x [0][1]
-9[LB] / 4[SB]	0	Speicherort von Variable x [1][0]
-8[LB] / 5[SB]	0	Speicherort von Variable x [1][1]
-7[LB] / 6[SB]	5	Argument von Funktion foo
-6[LB] / 7[SB]	0[SB]	Dynamic Link
-5[LB] / 8[SB]	16[CB]	Return Adresse von foo
-4[LB] / 9[SB]	5	Argument für Funktion bar m[0][0]
-3[LB] / 10[SB]	0	Argument für Funktion bar m[0][1]
-2[LB] / 11[SB]	0	Argument für Funktion bar m[1][0]
-1[LB] / 12[SB]	5	Argument für Funktion bar m[1][1]
0[LB] / 13[SB]	4[SB]	Dynamic Link
1[LB] / 14[SB]	12[CB]	Return Adresse von bar
2[LB] / 15[SB]	0	Rückgabewert von Funktion bar m[0][0]
3[LB] / 16[SB]	5	Rückgabewert von Funktion bar m[0][1]
4[LB] / 17[SB]	5	Rückgabewert von Funktion bar m[1][0]

Stackelement	Stack Inhalt	Bedeutung
5[LB] / 18[SB]	0	Rückgabewert von Funktion bar m[1][1]

## Aufgabe 3.3

a)

- Die Adresse des Zuweisungsziels für die Zuweisung **a[69] = 54** ist **71[LB]**.
- Die Adresse des Zuweisungsziels für die Zuweisung **b[5][8] = 27.5** ist **5 \* 11 + 9 + 130 = 194[LB]**

b)

- **Stackpositionen**
  - **v** = 2-17
  - **a** = 18
  - **r** = 19 - 29
  - **b** = 30

```

r@m[0][0] = 1 => 19[LB]
r@m[0][1] = 2 => 20[LB]
r@m[0][2] = 3 => 21[LB]
r@m[1][0] = 4 => 22[LB]
r@m[1][1] = 5 => 23[LB]
r@m[1][2] = 6 => 24[LB]
r@m[2][0] = 7 => 25[LB]
r@m[2][1] = 8 => 26[LB]
r@m[2][2] = 9 => 27[LB]
r@b = true => 28[LB]
r@i = 54 => 29[LB]
a = 80 => 18[LB]
r@b = false => 28[LB]

```

## Aufgab 3.4

- **missing (1):** `var int z;`
- **missing (2):** `val bool m = true;`
- **missing (3):** `e = mac(5, 13, 27);`
- **missing (4):** `RETURN (1) 3`
- **missing (5):** `ltl`
- **missing (6):** `CALL 12[CB]`
- **missing (7):** `JUMP 41[CB]`