

# Informatik Übungsstunde - Woche 3

## Negative Binärzahlen

In 4Bit Binary gilt  $1_{10} = 0001_2$  und  $0_{10} = 000_2$ .

→ Beobachtung:

$$x = -1 \Leftrightarrow \begin{array}{r} 0001 \\ + 1111 \\ \hline 0000 \end{array} \quad \leftarrow x$$

$$\Rightarrow -1_{10} = 1111_2$$

Wie berechnet man allg. binäre Darstellung von negativen Zahlen (hier im 4Bit System)?

Für  $x < 0$  führe folgende Schritte aus:

- 1) Binärdarstellung von  $|x|$  berechnen
- 2) Bits flippen
- 3) Die Zahl  $1_{10}$  dazurechnen

Bsp: Für  $x = -5$  gilt

$$1) |x| = 5_{10} = 0101_2$$

$$2) 0101 \xrightarrow{\text{flip bits}} 1010$$

$$3) 1010 \xrightarrow{+1_{10}} 1011 \rightarrow -5_{10} = 1011_2$$

Beobachtung: Das erste Bit gibt das Vorzeichen an!

Daraus folgt zudem im 4Bit System:

- Kleinste Zahl:  $-8_{10} = 1000_2$
- Grösste Zahl:  $7_{10} = 0111_2$



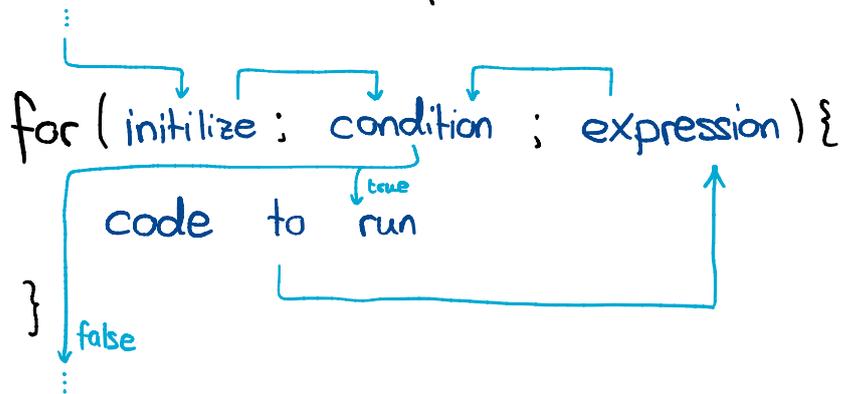
Bsp: Seien  $x, y$  beliebige Variablen vom Typ `int`.

•  $4 > 9/2 \parallel 5 == 19 \% 7 \parallel x++ == 7 \% 3 - y + 18 * 19 / 2$   
false true  $\rightarrow$  Short Circuit Evaluation  
 $\rightarrow$  true

•  $2 == 1 + 3 \ \&\& \ x / (9 - 3 * 3) == 4$   
false  $\rightarrow$  Short Circuit Evaluation  
 $\rightarrow$  false

## For-Loops:

Allgemeine For-Schleife:



Bsp:

```
for (int i = 0; i < 15; i = i + 5) {  
    std::cout << i << "\n";  
}
```

$\rightarrow$  Output: 0  
5  
10