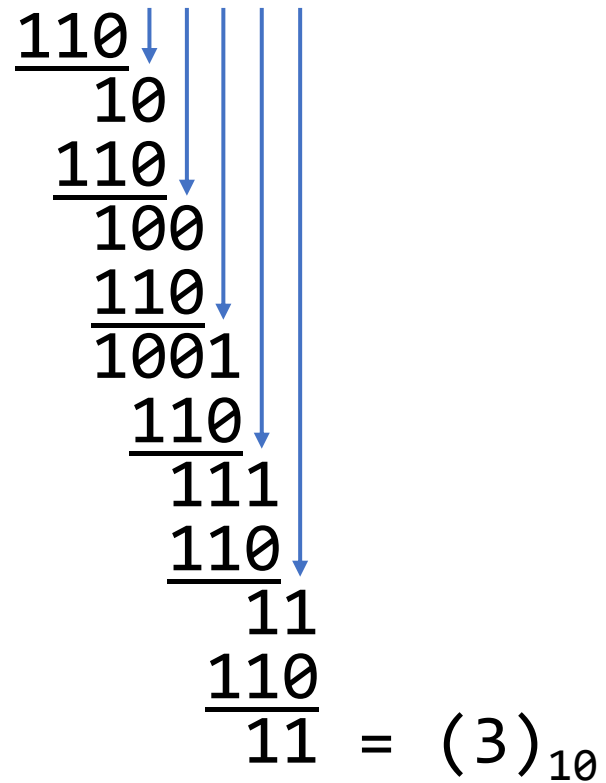


$$(1110 \ 0111)_2 = (231)_{10}$$

$$(0000 \ 0110)_2 = (6)_{10}$$

$$11100111 \div 110 = 100110 = (38)_{10}$$



$$\begin{array}{r}
 110 \\
 \underline{110} \\
 10 \\
 110 \\
 \underline{110} \\
 100 \\
 110 \\
 \underline{110} \\
 1001 \\
 110 \\
 \underline{110} \\
 111 \\
 110 \\
 \underline{110} \\
 11 \\
 110 \\
 \underline{110} \\
 11 = (3)_{10}
 \end{array}$$

```
neg := A < 0 ^ B < 0
```

```
if A < 0:  
    A := -A
```

```
if B < 0:  
    B := -B
```

```
Q := 0  
S := 7
```

```
while S >= 0: // Wiederhole, bis jeder Bit verarbeitet wurde.  
    if (A >> S) >= B: // Passt B in Teil von A?  
        Q := Q | (1 << S) // Setze Bit im Ergebnis an entsprechender Position.  
        A := A - (B << S) // Führe verschobene Subtraktion durch.  
    S := S - 1
```

```
if neg:  
    Q := -Q
```

```
return Q
```

Spezialregel für 1. Subtraktion

1. Iteration

$S := 7; A \gg 7 = 1; 1 \geq B$

$Q = 00000000$

2. Iteration

$S := 6; A \gg 6 = 11; 3 \geq B$

$Q = 00000000$

```
while S >= 0:
    if (A >> S) >= B:
        Q := Q | (1 << S)
        A := A - (B << S)
    S := S - 1
```

$A = (1110\ 0111)_2 = (231)_{10}$

$B = (0000\ 0110)_2 = (6)_{10}$

Durchführung 1. Subtraktion

3. Iteration

$S := 5$; $A \gg 5 = 111$; $7 \geq B$

```
while S >= 0:
    if (A >> S) >= B:
        Q := Q | (1 << S)
        A := A - (B << S)
    S := S - 1
```

Setze 6. Bit in Q auf 1.

$Q := 00100000$

$A = (1110\ 0111)_2 = (231)_{10}$

$B = (0000\ 0110)_2 = (6)_{10}$

Führe versetzte Subtraktion durch.

$B \ll 5 = 11000000$

$A - (B \ll 5) = 11100111 - 11000000$
 $= 00100111$

Weitere Subtraktionen

4. Iteration

$S := 4; A \gg 4 = 10; 2 \geq B$

$Q := 00100000$

5. Iteration

$S := 3; A \gg 3 = 100; 4 \geq B$

$Q := 00100000$

```
while S >= 0:
    if (A >> S) >= B:
        Q := Q | (1 << S)
        A := A - (B << S)
    S := S - 1
```

$A = (0010\ 0111)_2 = (39)_{10}$

$B = (0000\ 0110)_2 = (6)_{10}$

Weitere Subtraktionen

6. Iteration

$S := 2; A \gg 2 = 1001; 9 \geq B$

Setze 3. Bit in Q auf 1.

$Q := 00100100$

Führe versetzte Subtraktion durch.

$B \ll 2 = 00011000$

$A - (B \ll 5) = 00100111 - 00011000$
 $= 00001111$

```
while S >= 0:
    if (A >> S) >= B:
        Q := Q | (1 << S)
        A := A - (B << S)
    S := S - 1
```

$A = (0010\ 0111)_2 = (39)_{10}$

$B = (0000\ 0110)_2 = (6)_{10}$

Weitere Subtraktionen

7. Iteration

$S := 1; A \gg 1 = 111; 7 \geq B$

Setze 2. Bit in Q auf 1.

$Q := 00100110$

Führe versetzte Subtraktion durch.

$B \ll 1 = 00001100$

$A - (B \ll 5) = 00001111 - 00001100$
 $= 00000011$

```
while S >= 0:
    if (A >> S) >= B:
        Q := Q | (1 << S)
        A := A - (B << S)
    S := S - 1
```

$A = (0000\ 1111)_2 = (15)_{10}$

$B = (0000\ 0110)_2 = (6)_{10}$

Weitere Subtraktionen

8. Iteration

$S := 0; A \gg 0 = 11; 11 \geq B$

$Q := 00100110$

$S := -1 \rightarrow$ Schleife bricht ab.

$A = 00000011$ bleibt als Rest übrig.

$Q = 00100110$ ist der Quotient.

```
while S >= 0:
    if (A >> S) >= B:
        Q := Q | (1 << S)
        A := A - (B << S)
    S := S - 1
```

$A = (0000\ 0011)_2 = (3)_{10}$

$B = (0000\ 0110)_2 = (6)_{10}$