

Hintergrundinformation für dieses Aufgabenblatt: **Minimale_ALU.pdf**

12. Zeige exemplarisch, daß die logischen Verknüpfungen

- a) $a + b$
- b) $a \oplus b$
- c) $a \cdot (b + \bar{c})$

sich auf die Operationen NOT und AND zurückführen lassen.

13. Die „höheren“ Rechenarten Multiplikation, Potenzierung, (ganzzahlige) Division lassen sich durch geeignete Iteration auf die in der CPU als Hardware implementierten Grundoperationen „Addition“ und „Subtraktion“ zurückführen.

a) Nach Übergabe der ganzen Zahlen a und b liefert der Algorithmus **produkt** als **ergebnis** das Produkt $a * b$ iterativ, und zwar durch wiederholte Addition.

- α) Konzipiere den Algorithmus **produkt** als Struktogramm.
- β) Formuliere die Funktion **produkt** als Quelltext in Python.
- γ) Erstelle zur Verifikation eine Trace-Tabelle für $a=5, b=7$.

b) Nach Übergabe der ganzen Zahlen a und b liefert der Algorithmus **potenz** als **ergebnis** die Potenz $a ^ b$ iterativ, und zwar durch wiederholte Multiplikation.

- α) Konzipiere den Algorithmus **potenz** als Struktogramm.
- β) Formuliere die Funktion **potenz** als Quelltext in Python; dabei werde die Funktion **produkt** verwendet.
- γ) Erstelle zur Verifikation eine Trace-Tabelle für $a=2, b=5$.

c) Nach Übergabe der ganzen Zahlen a und b liefert der Algorithmus **quotient** als **ergebnis** den ganzzahligen Quotient $a // b$ iterativ, und zwar durch wiederholte Subtraktion.

- α) Konzipiere den Algorithmus **quotient** als Struktogramm.
- β) Formuliere die Funktion **quotient** als Quelltext in Python; dabei werde die Funktion **differenz** verwendet.
- γ) Erstelle zur Verifikation eine Trace-Tabelle für $a=59, b=12$.

14. Vervollständige den Quelltext zum Algorithmus „Grundrechenarten“, der nach Eingabe der als ganzzahlig vorausgesetzten Operanden **x** und **y** und nach Eingabe der Rechenoperation $(+, -, *, **, /)$ das verlangte Ergebnis liefert.

```

# Grundrechenarten
# Die "höheren" Rechenoperationen Multiplizieren, Potenzieren,
# Dividieren werden durch geeignete Iteration auf die
# Grundoperationen Addieren und Subtrahieren zurückgeführt.

def summe(a,b):
    return a + b

def differenz(a,b):
    return a - b

def produkt(a,b):
    .....
    .....

def potenz(a,b):
    .....
    .....

def quotient(a,b):
    .....
    .....

# Eingabe der Operanden
print ('Operanden:')
x = int(input('x = '))
y = int(input('y = '))
print()

# Eingabe der Rechenoperation
print('Operation:')
print('  Addition < + >')
print('  Subtraktion < - >')
print('  Multiplikation < * >')
print('  Division < / >')
print('  Potenz < ** >')
op = input()
print()

# Ausgabe des Resultats
if op == '+':
    print (x, ' + ', y, '=', summe(x,y))
elif op == '-':
    print (x, ' - ', y, '=', differenz(x,y))
elif op == '*':
    print (x, ' * ', y, '=', produkt(x,y))
elif op == '/':
    print (x, ' // ', y, '=', quotient(x,y))
elif op == '**':
    print (x, ' ^ ', y, '=', potenz(x,y))
else: print('falsche Eingabe')

```