

## Aufgabe 1:

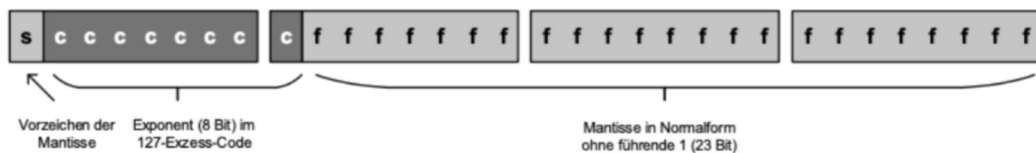
Fülle folgende Tabelle aus (Ohne Taschenrechner)

Hexadezimal	Binär	Dezimal
1024 <sub>hex</sub>	1000000100100	4132
D561	1101 0101 0110 0001 <sub>bin</sub>	54625
A410	1010010000010000	42000 <sub>dez</sub>
ABCD <sub>hex</sub>	1010101111001101	43981

## Aufgabe 2:

Geben Sie die folgenden Dezimalzahlen als 32-Bit Gleitkommazahlen nach IEEE-Format sowohl in binärer Form als auch in hexadezimaler Form an:

- a)  $1,67125 \times 10^2$
- b)  $-8,012225 \times 10^4$
- c)  $-4,0125 \times 10^3$



Im Folgenden bezeichnet  $d$  die Zahl in Dezimaldarstellung,  $b$  die Zahl in Binärdarstellung,  $e$  den Exponenten,  $c$  den Exponenten mit Bias und  $g$  die Zahl im 32-Bit IEEE Gleitkommaformat.

$$d = 1,67125 \cdot 10^2 = 167,125$$

$$b = 10100111,001_2 = 1,0100111001_2 \cdot 2^7$$

$$e = 7_{10}, c = 7_{10} + 127_{10} = 134_{10} = 10000110_2$$

$$g = 01000011001001110010000000000000 = 43272000_{16}$$

$$d = -8,012225 \cdot 10^4 = -80122,25$$

$$b = -1001100011111010,01_2 = -1,00110001111101001_2 \cdot 2^{16}$$

$$e = 16_{10}, c = 16_{10} + 127_{10} = 143_{10} = 10001111_2$$

$$g = 11000111100111000111110100100000 = C79C7D20_{16}$$

$$d = -4,0125 \cdot 10^3 = -4012,5$$

$$b = -111110101100,1_2 = -1,111101011001_2 \cdot 2^{11}$$

$$e = 11_{10}, c = 11_{10} + 127_{10} = 138_{10} = 10001010_2$$

$$g = 11000101011110101100100000000000 = C57AC800_{16}$$

---

### Aufgabe 3:

Führen Sie die angegebene logische Operation auf die binären Ganzzahlen durch (Ergebnis in binär)

Zahl 1:

0001 0001 v 1000 0100

10010101

Zahl 2:

0101 1010 XOR 1001 0101

1001110

---

### Aufgabe 4:

Finde sechs Fehler im Code (Sowohl Syntax als auch Logikfehler)

```
# Program to display the Fibonacci sequence up to n-th term

nterms = input("How many terms? ")          #int entfernt

# first two terms
n1, n2 = 0, 1                                #count = 0 variable muss definiert werden

# check if the number of terms is valid
if nterms >= 0:                               # muss kleiner gleich sein (logik fehler)
    print("Please enter a positive integer")
# if there is only one term, return n1
elif nterms == 1:                             #zwei gleich zeichen
    print("Fibonacci sequence upto",nterms,":")
    print("n1")                               #anführungszeichen entfernen
# generate fibonacci sequence
else:
    print("Fibonacci sequence:")
    while count < nterms:
        print(n1)
        nth = n1 + n2
        n1 = n2
        n2 = nth
        count + 1                             #muss += sein
```