

## Aufgabe 1: Zahlenbasis

Lösen die folgenden Umrechnungen mit erkennbarem Lösungsweg und kommentieren Sie ggf. Ihre Annahmen oder Probleme.

- a) Rechnen Sie die folgenden Zahlen vom Dezimalsystem ins Binär- und Hexadezimalsystem um:
- $113_{10}$
  - $257, 23_{10}$
- b) Rechnen Sie die folgenden Zahlen vom Binärsystem ins Dezimalsystem um:
- $101011010_2$
  - $10010, 1001_2$
- c) Rechnen Sie die folgenden Zahlen vom Hexadezimalsystem ins Dezimalsystem um:
- $E37A_{16}$
  - $39B, 2D8_{16}$
- d) Rechnen Sie folgende Binärzahl ohne Umweg über das Dezimalsystem direkt ins Hexadezimalsystem um.
- $0101101010110010111_2$
- e) Rechnen Sie die folgenden Zahlen ins Dezimalsystem um:
- $HODOR_{26}$
  - $STAR, WARS_{36}$

## Aufgabe 2: Collatz Conjecture

In dieser Aufgabe sollen Sie Programmflusskontrolle in Assembler erlernen. Machen Sie sich dazu mit Assembler-Sprüngen vertraut (Befehle: CMP, TEST, JMP, LOOP, *Jxx*). Wie hängen Sprünge mit den Flags im Register „rflags“ zusammen? Wann wird welches Flag im Register „rflags“ gesetzt? Was ist der Unterschied zwischen den Befehlen JG und JA?

Übersetzen Sie folgenden C-artigen Pseudocode in NASM-artigen Pseudocode:

```
if(x < 10) {  
    S();  
} else if(x == 10) {  
    T();  
} else if(x > 10) {  
    V();  
} else {  
    W();  
}
```

Finden Sie für das folgende Fragment zwei verschiedene, aber semantisch äquivalente Übersetzungen:

```
while(x != 10) {  
    x = S();  
}
```

**Collatz Conjecture** Implementieren Sie eine Funktion die, die Collatzfolge einer übergebenen Zahl bildet und die Länge der Folge zurück gibt.

Pseudocode:

```
collatz(n) {  
    k = 0;  
    while (n > 1) {  
        if (is_even(n)) {  
            n /= 2;  
        } else {  
            n *= 3;  
            n++;  
        }  
        k++;  
    }  
    return k;  
}
```

Die Funktion soll folgende Signatur haben:

```
uint64_t collatz(uint64_t n);
```

Ein geeigneter C-Wrapper wird ihnen gestellt.

*Hinweis: is\_even kann als  $n \bmod 2 = 0$  dargestellt werden.*

**Zusatzaufgabe** Übersetzen Sie auch noch die folgenden Pseudocodefragmente:

```
do {  
    x = S();  
} while (x != 10);
```

```
for (i = 0; i < 10; i++) {  
    S(i);  
}
```

```
x = S();  
switch (x) {  
case 1:  
    x += 100;  
case 2:  
    x -= 20;  
case 10:  
    x *= 10;  
default:  
    x = 100;  
}
```