

CT1 Übungsaufgaben

Datenübergabe / Schnittstelle zu Hochsprachen

Aufgabe 1

Gegeben ist folgender C-Code:

```
uint32_t logical_and(uint32_t a, uint32_t b, uint32_t c)
{
    return a & b & c;
}

int32_t main(void)
{
    uint32_t result;
    uint32_t x = 0x11223344;
    uint32_t y = 0xFFFF0000;
    uint32_t z = 0x33661122;

    result = logical_and(x, y, z);
}
```

Beim Start des Programmes wird die Variable x in R4, y in R5 und z in R6 abgelegt. Die Variable result wird in R7 abgelegt.

1. Welche Schritte führt der Caller (main) vor dem Aufruf der Funktion logical_and() durch? Wie werden die Parameter übergeben?

2. Wie gibt die Funktion logical_and() den Rückgabewert zurück?

3. Welche Operation führt der Call nach dem Aufruf der Funktion logical_and() durch?

Aufgabe 2

Gegeben ist das folgende C-Programm:

```
#include <utils_ctboard.h>

void swap_bad(int32_t c, int32_t d)
{
    /*WARNING: This code does not work*/
    int32_t temp = c;
    c = d;
    d = temp;
}

int32_t main(void)
{
    int32_t a = 3, b = 5;

    swap_bad(a,b);

    write_word(0x60000300, a);
}
```

- a. Die Swap-Funktion `swap_bad()` lässt sich ohne Fehler kompilieren und ausführen. Nach ihrem Aufruf in `main()` hat die Variable `a` jedoch immer noch den Wert 3. Erläutern Sie anhand der Calling Convention wo das Problem liegt.

- b. Schreiben Sie eine Funktion `swap_good()` die diesen Fehler behebt und korrekt funktioniert

Aufgabe 3

Gegeben sei das folgende C-Programm:

```
int32_t fakultaet_recursive(int32_t n)
{
    if(n < 2){
        return 1;
    }
    else{
        return n * fakultaet_recursive(n-1); // Break Point set here
    }
}

int32_t main(void)
{
    int32_t n = 20;

    int32_t result = fakultaet_recursive(n);
}
```

Das C-Programm berechnet die Fakultät von 20 und verwendet dazu Rekursion. Dabei ruft sich die Funktion selbst wieder auf. Dies wird so lange wiederholt, bis die Abbruchbedingung ($n < 2$) erreicht wird.

Wenn das Programm zum zweiten Mal beim Breakpoint in der Funktion fakultaet_recursive() gestoppt wird, ist das Stackframe der main()-Funktion und der aufgerufenen fakultaet_recursive()-Funktionen 16 Byte gross. Die Variable n wird in einem Register gespeichert.

Wie gross wird das Stackframe beim Ausführen des Programmes maximal?
