Karlsruher Institut für Technologie
Lehrstuhl für Programmierparadigmen
Sprachtechnologie und Compiler WS 2010/2011
Dozent: Prof. Dr.-Ing. G. Snelting
Übungsleiter: Matthias Braun matthias.braun@kit.edu

Übungsblatt 9 Ausgabe: 18.06.2012 Besprechung: 25.06.2012

Aufgabe 1: *Aufrufkonventionen*

Beschreiben Sie die Aufraufkonventionen (Calling Conventions) von 2 der folgenden Prozessorarchitekturen auf einem Betriebssystem Ihrer Wahl:

1. IA-32
2. PowerPC
3. AMD-64/Intel 64
4. ARM
5. MIPS
6. Sparc (V8)

Suchen Sie hierzu im Internet nach den entsprechenden Dokumenten im Internet (z.B. auf den Herstellerseiten).

Geben Sie für jede der von Ihnen gewählten Plattformen an, wie die Argumente beim Aufruf folgender C-Funktion übergeben werden. Wie wird der Rückgabewert zurück gegeben?

```
struct test_struct {
    int x;
    int y;
};

int foobar(int i, char c, double d, int j, struct test_struct s) {
    return 42;
}
```

Aufgabe 2: *Activation Records*

Gegeben folgendes Programm:

```
int iterations;

int fib(int x)
{
    int r;

    iterations++;
    if (x == 0) {
        r = 0;
    } else if (x == 1) {
        r = 1;
    } else {
        r = fib(x-1) + fib(x-2);
    }
    return r;
```

```
}

int main(void)
{
    int input;
    scanf("%d", &input);
    printf("Result: %d, needed %d iterations\n", fib(input), iterations );
    return 0;
}
```

Stellen Sie sich einen nicht optimierenden Compiler vor, der alle Variablen auf dem Stack ablegt. Außerdem verlangt die Aufrufkonvention dass alle Argumente auf dem Stack übergeben werden. Der Prozessor selbst legt beim Betreten einer Funktionen die Rücksprungadresse auf den Stack.

Stellen Sie die Struktur des Stacks unmittelbar vor dem Aufruf der Funktion fib dar und nach dem 2. Rekursionsschritt der Funktion fib.

Aufgabe 3: *Array-Adressierung*

Geben Sie für ein 4-dimensionales Array $a[-10..10, -20..20, 30..42, 0..100]$ von 32-Bit Ganzzahlen den erzeugten Zwischencode (Bytecode oder Tripelcode) zur Adressberechnung gemäß Hornerschema an. Berechnen Sie zuvor die korrekte virtuelle Anfangsadresse $adr(a[0,0,0,0])$. Sie dürfen annehmen, dass die l_μ Compilezeitkonstanten sind.