

Tutoriumsblatt 5

Rechnerarchitektur im Sommersemester 2023

Zu den Modulen C, D, G

Besprechung: 22.05.23 bis 26.05.23

Aufgabe 1: (T) Parameterübergabe bei Unterprogrammaufrufen (- Pkt.)

Für die Parameterübergabe bei Prozeduraufrufen existieren verschiedene Möglichkeiten.

- Erläutern Sie zunächst die Begriffe *call by value* und *call by reference*. Geben Sie zu beiden Konzepten je ein Beispiel in einer Hochsprache an.
- Schreiben Sie nun ein SPIM-Programm, das den Durchschnitt der Werte eines Feldes berechnet. Die Berechnung selbst soll dabei ein Unterprogramm erledigen. Die Übergabe des Feldes soll nach dem Konzept *call by value* erfolgen.

Achtung: Das Hauptprogramm soll dem Unterprogramm **alle** zur Berechnung notwendigen Werte über den Stack zur Verfügung stellen! Sie dürfen bei Ihrer Implementierung davon ausgehen, dass sich das Feld bereits im Speicher befindet.

- Schreiben Sie Ihr Programm aus Aufgabe b) so um, dass die Übergabe des Feldes nach dem Konzept *call by reference* funktioniert.

Achtung: Das Hauptprogramm soll dem Unterprogramm **ausschließlich** Speicheradressen zur Berechnung zur Verfügung stellen! Sie dürfen wieder davon ausgehen, dass sich das Feld bereits im Speicher befindet. Sie dürfen zur Übergabe der Adressen an das Unterprogramm die laut Konvention dafür vorgesehenen Register \$a0 - \$a3 verwenden. Das Ergebnis des Unterprogrammaufrufes dürfen Sie dem Hauptprogramm über das Register \$v0 zur Verfügung stellen.

Aufgabe 2: (T) SPIM: Fakultät iterativ und rekursiv (- Pkt.)

In dieser Aufgabe betrachten wir verschiedene Möglichkeiten, die Fakultätsfunktion zu implementieren.

- Schreiben Sie ein SPIM-Programm, das die Fakultätsfunktion nach folgendem **iterativen** Algorithmus berechnet:

```
1 int fak(int n) {  
2     if(n < 0)  
3         return -1;  
4     int result = 1;  
5     while(n > 0) {
```

```
6         result = result * n;
7         n = n - 1;
8     }
9     return result;
10 }
```

Gehen Sie bei Ihrer Implementierung davon aus, dass der auf der Website bereitgestellte Programmrahmen `fak_it_rec_rahmen.s` bereits existiert. Das Ergebnis der Berechnung soll am Ende auf der Konsole ausgegeben werden.

- b. Schreiben Sie nun ein SPIM-Programm, das die Fakultätsfunktion nach folgendem **rekursiven** Algorithmus berechnet:

```
1 int fak(int n) {
2     if(n < 0)
3         return -1;
4     if(n == 0)
5         return 1;
6     else
7         return n * fak(n-1);
8 }
```

Gehen Sie bei Ihrer Implementierung davon aus, dass der auf der Website bereitgestellte Programmrahmen `fak_it_rec_rahmen.s` bereits existiert. Implementieren Sie den rekursiven Algorithmus mit Hilfe einer Prozedur. Achten Sie dabei insbesondere auf eine korrekte Realisierung des Prozeduraufrufs nach den MIPS-Prozedur-Konventionen! Das Ergebnis der Berechnung soll am Ende wieder auf der Konsole ausgegeben werden.

Hinweis: Diese Aufgabe wird **nicht** in den Tutorienvideos, sondern nur diese Woche (22.05.23 bis 26.05.23) in den Tutorien besprochen.