
Übungsblatt 2

Abgabe: 07.11.2022 9:00 Uhr

- Dieses Übungsblatt muss im Team abgegeben werden (Einzelabgaben sind nicht erlaubt!).
- Die **Zeitangaben** geben zur Orientierung an, wie viel Zeit für eine Aufgabe später in der Klausur vorgesehen wäre; gehen Sie davon aus, dass Sie zum jetzigen Zeitpunkt wesentlich länger brauchen und die angegebene Zeit erst nach ausreichender Übung erreichen.

* leichte Aufgabe / ** mittelschwere Aufgabe / *** schwere Aufgabe

Aufgabe 5 (*Fetch/Decode/Execute/Write-Back*)

In dieser Aufgabe sollen Sie den Fetch/Decode/Execute/Write-Back-Zyklus für die Abarbeitung verschiedener Maschinenbefehle beschreiben.

Orientieren Sie sich dabei an Beispiel 2.12 im Foliensatz. Geben Sie also insbesondere alle relevanten Register und deren jeweilige Belegung an, verzichten Sie aber auf die Beschreibung von Steuersignalen.

Betrachten Sie dazu das folgende Maschinenprogramm:

```
P1: INIT S2
P2: SPRUNG0 P6, S1
P3: ADD S2, S1
P4: DEKREMENT S1
P5: SPRUNG P2
P6: RÜCKGABE S2
```

- a) (*, 6 Minuten)
Beschreiben Sie die Abarbeitung des Maschinenbefehls an Adresse P1.
- b) (**, 6 Minuten)
Beschreiben Sie die Abarbeitung des Maschinenbefehls an Adresse P3.
- c) (***, 6 Minuten)
Beschreiben Sie die Abarbeitung des Maschinenbefehls an Adresse P2.

Aufgabe 6 (*Kommunikation zwischen Steuerwerk und Speicherwerk*)

In dieser Aufgabe sollen Sie detailliert die Kommunikation zwischen Steuerwerk und Speicherwerk bei Schreib- und Lesevorgängen beschreiben.

Orientieren Sie sich dabei an Beispiel 2.13 im Skript. Geben Sie dabei insbesondere alle Steuer-signale und alle relevanten Register und deren jeweilige Belegung an.

Betrachten Sie dazu das folgende Maschinenprogramm:

```
P1: INIT S2
P2: SPRUNG0 P6, S1
P3: ADD S2, S1
P4: DEKREMENT S1
P5: SPRUNG P2
P6: RÜCKGABE S2
```

- a) (**, 6 Minuten)
Beschreiben Sie detailliert die Kommunikation zwischen Steuerwerk und Speicherwerk beim Lesen des Operanden im Decode-Schritt der Abarbeitung des Maschinenbefehls an Adresse P6.
- b) (**, 6 Minuten)
Beschreiben Sie detailliert die Kommunikation zwischen Steuerwerk und Speicherwerk beim Lesen des Befehls im Fetch-Schritt der Abarbeitung des Maschinenbefehls an Adresse P1.
- c) (*, 6 Minuten)
Beschreiben Sie detailliert die Kommunikation zwischen Steuerwerk und Speicherwerk beim Schreiben des Ergebnisses im Write-Back-Schritt der Abarbeitung des Maschinenbefehls an Adresse P3.

Aufgabe 7 (*Maschinenprogramme erstellen*)

Für alle Maschinenprogramme in dieser Aufgabe gelten folgende Vorgaben aus dem Skript:

- Es sind **ausschließlich** die Maschinenbefehle aus dem Skript zulässig (Definitionen 2.17 und 2.18).
- Beachten Sie, dass die Maschinenbefehle lediglich Adressen als Operanden annehmen. Für konstante Werte verwenden Sie den Datenteil.
- Für Adressen sind die Notationen aus dem Skript einzuhalten (Definition 2.16), insbesondere: Der erste Befehl des Programms liegt an Adresse P1, der Datenteil beginnt mit Adresse D1 und der Stack Frame mit Adresse S1.

- a) (*, 4 Minuten)
Beschreiben Sie das Ergebnis des folgenden Maschinenprogramms, wenn der Wert an S1 eine nicht-negative ganze Zahl ist.

```
P1: INIT S2
P2: SPRUNG0 P6, S1
P3: ADD S2, S1
P4: DEKREMENT S1
P5: SPRUNG P2
P6: RÜCKGABE S2
```

b) (**, 10 Minuten)

Erstellen Sie ein Maschinenprogramm, das die folgende C-Funktion realisiert. Beschreiben Sie außerdem in wenigen Sätzen die Vorgehensweise Ihres Maschinenprogramms und geben Sie dazu an, welche Adressen Sie für Eingabeparameter und lokale Variablen benutzen.

```
int is_five_times_plus_one(int a, int b)
{
    return a * 5 + 1 - b == 0;
}
```

c) (***, 8 Minuten)

Sei an S1 eine nicht-negative ganze Zahl gespeichert. Erstellen Sie ein Maschinenprogramm, das 1 zurückgibt, falls die an S1 gespeicherte Zahl interpretiert als ASCII-Zeichen einer arabischen Ziffer entspricht, und sonst 0 zurückgibt. Beschreiben Sie außerdem in wenigen Sätzen die Vorgehensweise Ihres Maschinenprogramms

Aufgabe 8 (*C-Funktion zu Maschinenprogramm finden*)

In dieser Teilaufgabe ist jeweils ein Maschinenprogramm vorgegeben. Sie sollen eine C-Funktion erstellen, die durch dieses Maschinenprogramm realisiert wird. Geben Sie dazu jeweils an, welche Konstanten, Eingabeparameter und lokale Variablen Sie den verwendeten Adressen zuordnen. Geben Sie außerdem der C-Funktion einen Namen, der die von ihr (bzw. die vom Maschinenprogramm) ausgeführte Aufgabe beschreibt.

a) (**, 6 Minuten)

```
P1: INIT S2
P2: SPRUNG0 P5, S1
P3: SPRUNGNEG P6, S1
P4: INKREMENT S2
P5: RÜCKGABE S2
P6: DEKREMENT S2
P7: RÜCKGABE S2
```

b) (**, 6 Minuten)

```
P1: INIT S3
P2: ADD S3, S1
P3: SUB S3, S2
P4: SPRUNGNEG P7, S3
P5: SUB S1, S2
P6: SPRUNG P1
P7: RÜCKGABE S1
```

Hinweis: Gehen Sie davon aus, dass der Wert an S1 nicht-negativ und der Wert an S2 positiv ist.

c) (**, 8 Minuten)

```
P1: INIT S2
P2: INIT S3
P3: ADD S3, D1
P4: SPRUNG0 P9, S1
P5: DEKREMENT S1
P6: ADD S2, S3
P7: ADD S3, D1
P8: SPRUNG P4
P9: RÜCKGABE S2
D1: 2
```