

Die John-von-Neumann-Architektur mit Johnny entdecken

Material für den Unterricht der Sekundarstufe II (Grund- und Leistungskurs)

Autor: Dirk Schwenn
neue friedländer gesamtschule
dirk.schwenn@web.de

Überarbeitung: Tino Hempel, 2020

Name: _____

Das Material darf unter der folgenden Lizenz verwendet werden:



Johnny 1: Lineare Programme

Video	Vorkenntnisse	Hilfsmittel
1_Linear.mp4	<ul style="list-style-type: none"> Grundlagen der Arbeitsweise mit MOPS (Klasse 10) Aufbau und Bestandteile mithilfe des LB S. 234 ff. 	<ul style="list-style-type: none"> Dokumentation Johnny, S. 7 bis 10 Befehlssatzes Johnny Datei: 01_1_Johnny.ram Datei: 01_6_Nachfolgersumme.ram

- Informieren Sie sich im Video über die Bedienung des Simulators *Johnny*. Bearbeiten Sie parallel die Aufgabe 2.
- Implementieren und testen Sie das Programm aus dem Video.

```
000: TAKE 004
001: ADD 005
002: SAVE 006
003: HLT 000
004:      012
005:      008
```

Was leistete das Programm prinzipiell?

Geben Sie die Adresse der Speicherzelle der Operanden an.

Begründen Sie, dass die Adresse 006 die Speicherzelle für das Ergebnis ist.

- Ermitteln Sie mithilfe des Befehlssatzes die Wirkung folgender Befehle:

TAKE

SAVE

ADD
SUB

INC
DEC

HLT

- Ändern Sie den Befehl auf der Adresse 001 in SUB 005. Testen Sie das modifizierte Programm. Ermitteln Sie den berechneten Term sowie das erwartete und das berechnete Ergebnis.

Term	erwartetes Ergebnis	berechnetes Ergebnis

- Modifizieren Sie das Programm so, dass die Operanden vertauscht werden. Ermitteln Sie den berechneten Term sowie das erwartete und das berechnete Ergebnis. Leiten Sie eine Schlussfolgerung ab.

Term	erwartetes Ergebnis	berechnetes Ergebnis

Schlussfolgerung:

- 6) Begründen Sie, dass das nachfolgende Programm 01_6_Nachfolgersumme.ram die Summe aus dem in der Zelle 020 gespeicherten Wert und seinem berechneten Nachfolger bestimmt.

000: TAKE 020 Begründung:
001: SAVE 021
002: INC 021
003: ADD 020
004: SAVE 021
005: HLT 000
020: 007

- 7) Modifizieren Sie das Programm aus der vorherigen Aufgabe so, dass es die Summe aus dem in der Zelle 020 gespeicherten Wert und seinem übernächsten Nachfolger bestimmt. Das Ergebnis soll in der Zelle 021 abgespeichert werden.

Adresse	Befehl	Operand

- 8) Nur für LK: Entwickeln Sie ein Programm, das die Summe aus dem in der Zelle 020 gespeicherten Wert, seinem Vorgänger und seinem Nachfolger bestimmt. Das Ergebnis soll in der Zelle 021 abgespeichert werden.

Adresse	Befehl	Operand

Johnny 2: Sprunganweisungen

Video	Vorkenntnisse	Hilfsmittel
2_Sprung.mp4	<ul style="list-style-type: none"> Johnny 1 	<ul style="list-style-type: none"> Dokumentation Johnny, S. 7 bis 10 Befehlssatzes Johnny Datei: 02_1_max.ram

- 1) Informieren Sie sich im Video über die Wirkung von Sprunganweisungen in *Johnny*. Bearbeiten Sie parallel die Aufgabe 2.
- 2) Implementieren und testen Sie das Programm aus dem Video, das zur Bestimmung des Maximums der Inhalte zweier Speicherzellen dient.

```

000: TAKE 017
001: SUB 018
002: SAVE 021
003: TST 021
004: JMP 008
005: TAKE 018
006: SAVE 020
007: HLT 000
008: TAKE 017
009: SAVE 020
010: HLT 000
017: 055
018: 093

```

Begründen Sie, dass die Adresse 020 die Speicherzelle für das Ergebnis ist.

- 3) Ermitteln Sie mithilfe des Befehlssatzes die Wirkung folgenden Befehle:

TST

JMP

- 4) Testen Sie das Programm mit der Bedingung, dass in 017 ein Wert steht, der

- a) größer als der in 018 ist.
- b) kleiner als der in 018 ist.
- c) identisch mit dem in 018 ist.

Nutzen Sie die nebenstehende Tabelle. Leiten Sie eine Schlussfolgerung ab.

Schlussfolgerung:

	Zelleninhalt		
	vorher		nachher
	017	018	020
a)			
b)			
c)			

6) Modifizieren Sie das Programm so, dass das Minimum der Inhalte der Zellen 017 und 018 in der Zelle 020 gespeichert wird.

Adresse	Befehl	Operand
000		
001		
002		
003		
004		
005		
006		
007		
008		
009		
010		
011		
012		
013		
014		
015		
016		
017		
018		
019		
020		
021		

7) Nur für LK: Modifizieren Sie das Programm so, dass das Maximum der Inhalte in den Zellen 017, 018 und 019 in der Zelle 020 gespeichert wird.

Adresse	Befehl	Operand
000		
001		
002		
003		
004		
005		
006		
007		
008		
009		
010		
011		
012		
013		
014		
015		
016		
017		
018		
019		
020		
021		

Johnny 3: Wiederholungen in der Programmabarbeitung

Video	Vorkenntnisse	Hilfsmittel
3_Schleifen.mp4	<ul style="list-style-type: none"> • Johnny 2 	<ul style="list-style-type: none"> • Dokumentation Johnny, S. 7 bis 10 • Befehlssatzes Johnny • Datei: 03_4_division.ram • Datei: 03_6_test.ram

- 1) Informieren Sie sich im Video über die Programmierung von Wiederholungen in *Johnny*. Bearbeiten Sie parallel die Aufgabe 2.
- 2) Implementieren Sie das Programm. Protokollieren Sie die Arbeitsweise des Programms zur Bestimmung des Produkts der Werte in den Speicherzellen 010 und 011 in Zelle 012. Schalten Sie dazu die Sicht auf das Steuerwerk frei.

```

000: NULL 012
001: TAKE 012
002: ADD 010
003: SAVE 012
004: DEC 011
005: TST 011
006: JMP 001
007: HLT 000

```

Programmcounter (in der Form 000 → 001 → 002 → ...):

Speicher- und Registerbelegung:

010:	5												
011:	3												
012:	0												
Akku:													

- 3) Testen Sie das Programm mit verschiedenen Werten einschließlich Null. Leiten Sie Schlussfolgerungen ab.

- 4) Analysieren Sie das Programm 03_4_division.ram. Die Speicherzelle 018 beinhaltet den Dividenten, die Zelle 019 den Divisor. Das Ergebnis wird in Zelle 020 abgelegt. Protokollieren Sie den Programmablauf. Beschreiben Sie die prinzipielle Arbeitsweise des Algorithmus.

```

000: NULL 020
001: TAKE 018
002: SUB 019
003: SAVE 018
004: INC 020
005: TST 018
006: JMP 001
007: HLT 000
018:      006
019:      002
020:      000

```

Programmcounter (in der Form 000 → 001 → 002 → ...):

Speicher- und Registerbelegung:

018:	6												
019:	2												
020:	0												
Akku:													

- 5) Testen Sie das obige Programm für die Berechnung von 21:7, 22:7 und 20:7. Beschreiben Sie Ihre Beobachtung. Leiten Sie Schlussfolgerungen ab.

- 6) Nur für LK: In der Datei 03_6_test.ram befinden sich in den Zellen 000 bis 017 der Algorithmus. Ab der Zelle 021 können fortlaufend Daten größer Null eingetragen werden. Eine Null beendet die Datenfolge. Das Ergebnis legt der Algorithmus in der Zelle 020 ab.

```

000: TAKE 021
001: SAVE 020
002: TAKE 021
003: SUB 020
004: SAVE 019
005: TST 019
006: JMP 008
007: JMP 010
008: TAKE 021
009: SAVE 020
010: INC 002
011: INC 008
012: INC 013
013: TST 021
014: JMP 002
015: HLT 000
016: TAKE 020
017: HLT 000
018:
019:
020:
021:      009
022:      007
023:      025

```

- a) Protokollieren Sie den Programmlauf für die gegebenen Werte. Beachten Sie dabei, dass sich der Quelltext des Programms während der Laufzeit ändert! Nutzen Sie eine Speicherbelegungsübersicht der Zellen 002, 008, 013 und 020.

002:	21												
008:	21												
013:	21												
020:													

- b) Testen Sie das Programm für verschiedene Datenreihen. Beachten Sie dabei, dass das Programm für jede Datenreihe neu zu laden und die Daten ab Zelle 021 anzupassen sind. Beschreiben Sie die Wirkung des Programms.

Johnny 4: Von-Neumann-Zyklus und Erweiterungsvorschläge

Video	Vorkenntnisse	Hilfsmittel
4_Zyklus.mp4	<ul style="list-style-type: none">• Johnny 2 (GK)• Johnny 3 (LK)	<ul style="list-style-type: none">• Dokumentation Johnny S. 7 bis 10• Befehlssatzes Johnny

- 1) Informieren Sie sich im Video über den von-Neumann-Zyklus.
Stellen Sie den Zyklus grafisch dar. Erfassen Sie dabei die genauen Aufgaben in den Phasen. Nutzen Sie Ihre Kenntnisse aus den vorherigen Lektionen zum Steuerwerk.

- 2) Nur für LK: Dokumentieren Sie den Programmablauf für ein Programm aus Johnny 2.
- 3) Nur für LK: Bearbeiten Sie zwei von drei Aufgaben.
 - a) Erweitern Sie das Programm `03_2_multiplikation.ram` so, dass es stets korrekt und möglichst effizient rechnet.
 - b) Erweitern Sie das Programm `03_4_division.ram` so, dass es stets korrekt rechnet und ggf. zusätzlich den Rest der Division bestimmt.
 - c) Implementieren Sie ein Programm zur Berechnung der Potenz a^2 .