

Grundbegriffe der Informatik — Aufgabenblatt 8

Matr.nr.:

Nachname:

Vorname:

Tutorium Nr.: Tutor*in:

Ausgabe: 10. Dezember 2021, 12:00 Uhr

Abgabe: 17. Dezember 2021, 12:30 Uhr
in dem Holzkasten neben Raum -119
im UG des Info-Gebäudes (50.34)

Lösungen werden nur korrigiert, wenn sie

- handschriftlich erstellt sind (Tablet-Ausdruck erlaubt) und
 - mit dieser Seite als Deckblatt
 - in der oberen **linken** Ecke zusammengeheftet
- rechtzeitig** abgegeben werden.

Abgaberegeln für Teilnehmer der Online-Tutorien:

- handschriftlich erstellt (lesbare Fotos akzeptiert)
- **rechtzeitig**, mit diesem Deckblatt in **genau einer** PDF-Datei
- direkt an den entsprechenden Tutor abgeben.

*Von Tutor*in auszufüllen:*

erreichte Punkte

Blatt 8: / 20

Blätter 7 – 8: / 40 [+3]

Gehen Sie in allen Aufgaben vom MIMA-Befehlssatz aus dem Skript (S. 91 - 94) aus. Beim Erstellen von eigenen Programmen dürfen Sie zusätzliche Adressen verwenden. Geben Sie diesen aussagekräftige Namen!

Aufgabe 8.1 (1,5 + 1 + 1,5 + 1 = 5 Punkte)

Rechts sehen Sie ein MIMA-Programm.

buffer und *k-buffer* sind zwei beliebige Adressen.

- a) Berechnen Sie jeweils den Wert des *Akkumulators*, wenn das Programm, mit den jeweiligen Werten im *Akkumulator*, bzw. an Adresse *k* aufgerufen wird:

<i>Akku</i>	<i>k</i>
i) 0111	0100
ii) 0110	0010
iii) 0000	0011

Anmerkung: Um Ihnen Arbeit zu sparen tun wir hier so, als ob die MIMA im 4-Bit-Zweierkomplement rechne.

- b) Was berechnet das Programm allgemein für Start-*Akku*-werte ≥ 0 sowie *k*-Werte > 1 ?

- c) Bei den Berechnungen für a) ist Ihnen bestimmt aufgefallen, dass das Programm „ineffizient“ ist, weil bestimmte Befehle unnötig häufig wiederholt werden. Modifizieren Sie das Programm, sodass dies nicht mehr der Fall ist. Die Ergebnisse des Programms sollen dabei unverändert bleiben.

- d) Das Programm „funktioniert nicht“ falls im *Akkumulator* zu Beginn Werte < 0 stehen. Modifizieren Sie das (Ursprungs-)Programm, sodass es für *Akku*-Werte < 0 analog wie für *Akku*-Werte ≥ 0 funktioniert.

Heben Sie bei c) und d) Ihre Änderungen zum Ursprungsprogramm (z.b.) farblich hervor. Achten Sie auf lesbare Programme.

Anfang:	STV	<i>buffer</i>
	LDV	<i>k</i>
	NOT	
	STV	<i>k-buffer</i>
	LDC	1
	ADD	<i>k-buffer</i>
	ADD	<i>buffer</i>
	JMN	Ende
	JMP	Anfang
Ende:	ADD	<i>k</i>
	HALT	

Aufgabe 8.2 (0,5 + 1,5 = 2 Punkte)

Irgendwie hat sich die Lieferung von Wunsch.de als Reinflall entpuppt. Statt „mega intelligenten maschinellen Assistenten“ (mimAs) wurden Ihnen „minimal machines“ (MIMAs) geliefert. Diese Geräte sind in der Praxis deutlich weniger hilfreich als es hyperintelligente-Quanten-KIs gewesen wären. Insbesondere der sehr limitierte Befehlssatz macht Programmierung dafür etwas mühselig*. Glücklicherweise können Sie mit Ihrem jüngst erworbenen Verständnis von Prozessoren den Befehlssatz durch eigene kleine MIMA-Programme geschickt erweitern. Beenden Sie ihre Programme **nicht** mit HALT. In nachfolgenden (Teil-)Aufgaben dieses Übungsblattes dürfen Sie die hier eingeführten Programme als zusätzliche MIMA-Befehle verwenden.

- a) INC *addr*

soll den Wert an Adresse *addr* inkrementieren, also um eins erhöhen. Das Ergebnis soll wieder in *addr* gespeichert werden.

- b) SUBI *addr*

soll den Wert, der sich an genau der Adresse befindet, die selbst in *addr* gespeichert ist, vom aktuellen *Akkumulator*-Inhalt subtrahieren. Das Ergebnis soll im *Akkumulator* stehen.

Hinweis: $a - b$ ist immer noch $a + (-b)$ // und LA immer noch wichtig ;-)

*Dr. Meta hat entscheidend Wichtigeres zu tun, weshalb er diese Aufgabe natürlich Ihnen anvertraut.

Aufgabe 8.3 (5 Punkte)

Während Sie **INC** *addr* und **SUBI** *addr* programmiert haben, hat jemand schnell noch einen Befehl **MOD** *addr* programmiert, mit dem der aktuell im *Akkumulator* befindliche Wert **mod** dem Wert an Adresse *addr* gerechnet werden kann. Das Ergebnis dieser Rechnung steht wieder im *Akkumulator*.

Mit Hilfe dieser zusätzlichen Befehle können Sie nun ein Programm schreiben, das die B.I.R.D.-Daten automatisiert entschlüsselt. Die verschlüsselte Nachricht sowie das geeignete Passwort liegen als *Liste* von Zeichen jeweils in zusammenhängenden Adressbereichen. Die Zeichen sind dabei praktischerweise als Zahlenwert im Zweierkomplement gespeichert, sodass sie diese Werte einfach verrechnen können. Für ihr Programm kennen Sie 4 Adressen des Datenspeichers: *m-start*, *p-start*, bzw. *m-end*, *p-end*, an denen jeweils das erste, bzw. letzte Zeichen von Nachricht und Passwort gespeichert ist. Zum Entschlüsseln, sollen Sie die Zeichen jeweils einzeln verrechnen, sodass gilt: $e_i = (m_i - p_{i^*}) \bmod k$, wobei e_i, m_i, p_{i^*} je das *i*-te Zeichen der entschlüsselten Nachricht, der verschlüsselten Nachricht, bzw. des Passwortes* ist. Beachten Sie, dass das Passwort unter Umständen kürzer als die verschlüsselte Nachricht ist. In diesem Fall müssen Sie wiederholt über das Passwort iterieren (deshalb der *).

Schreiben Sie die entschlüsselte Nachricht wieder an die selben Adressen der ursprünglichen Nachricht, also *m-start* bis inkl. *m-end*. Der Wert *k* für die Modulorechnung liegt dem Programm ebenfalls an Adresse $\$k$ vor. **Achten Sie auf ein lesbares Programm!** Nummerieren Sie die einzelnen Zeilen und beschreiben Sie außerdem kurz die Funktion einzelner Programmblöcke, sodass insgesamt Ihr gesamtes Programm beschrieben ist.

Hinweis: Orientieren Sie sich zur Verarbeitung einer Liste mittels indirekter Adressierung an den VL-Folien zum Aufsummieren von Elementen (Folie 61-63)

Aufgabe 8.4 (1 + 1 + 3 + 1 + 2 = 8 Punkte)

Gegeben eine kontextfreie Grammatik $G = (N, T, S, P)$, mit:

- $N = \{S, A, B, C\}$
- $T = \{a, b\}$
- $P = \{S \rightarrow aS|bS|aA, A \rightarrow aA|bA|aB, B \rightarrow aB|bB|aC, C \rightarrow aC|bC|\varepsilon\}$

- Geben Sie zwei verschiedene Ableitungsbäume für das Wort $w = abaaba$ an.
- Geben Sie eine formale Definition der von G erzeugten Sprache $L(G)$ ohne Verwendung natürlicher Sprache an. (Teilpunkte bei Verwendung natürlicher Sprache).
- Sei $w = aabaa$. Wie viele verschiedene Ableitungsbäume für die Erzeugung von w durch G gibt es? Beweisen Sie.

Hinweis: Argumentieren Sie über die Struktur der Produktionsmenge

Hinweis: Welche „vier Zwischenphasen“ durchläuft ein Wort bei der Erzeugung durch G ?

- Welches $w \in (\{a, b\}^5 \cap L(G))$ hat die meisten verschiedenen Ableitungsbäume? (Angabe genügt)
- Definieren Sie eine Grammatik $G' = (N', T', S', P')$, mit $L(G') = L(G)$ und minimalem $|N'|$.