

Übungen

Grundsätzliches:

- Die Aufgaben dienen der Festigung und Vertiefung der in der Vorlesung vermittelten Lehrinhalte.
- Es ist nicht erforderlich, alle Aufgaben zu lösen.
Im Vordergrund sollte stets eine Problemdiskussion stehen.
- Alternative Aufgaben können von den Studenten vorgeschlagen werden.
- Zur Klausur wird die Kompetenz zur Lösung vergleichbarer Aufgaben erwartet.

Übung 1 (Einführung)

1. Steuern und Regeln
 - a. Was ist der Unterschied zwischen Steuern und Regeln?
 - b. Welche Aufgaben haben Sensoren und Actoren?
2. Information
 - a. Wie hoch ist der Informationsgehalt einer Nachricht
„Die gesuchte Person ist ein Mann“ (Annahme: 1/2 m; 1/2 w)
 - b. Wie hoch ist der Informationsgehalt einer Nachricht
„Die gesuchte Person ist Linkshänder“ (1/16 li; 15/16 re)
 - c. Wie hoch ist der Informationsgehalt einer Nachricht
„Die gesuchte Person ist Rechtshänder“ (1/16 li; 15/16 re)
3. Welche Geräte verarbeiten Information analog bzw. digital ?
 - a. Geldautomat
 - b. Lautsprecher
4. Zahlendarstellung
 - a. Stellen Sie die Dezimalzahlen „141“ und „76“ als Binärzahl dar.
 - b. Stellen Sie o.g. Zahlen hexadezimal dar.
5. Sind Computer für folgende Anwendungen nutzbar, gibt es evtl. Beschränkungen?
 - a. Mathematische Berechnungen beliebiger Art
 - b. Steuerung von Produktionsanlagen
 - c. Automatisierung des Aktienhandels
6. Was ist Hard- bzw. Software?
 - a. Computermaus
 - b. Betriebssystem
 - c. Drucker
7. Diskutieren Sie ein Beispiel für
 - a. Schwere Auswirkung von Rechner- bzw. Rechnernetzausfällen
 - b. Computerkriminalität

Übung 2 (Rechnerarchitektur)

1. Erkundigen Sie sich bei Ihren (Groß-)Eltern, alternativ bei Wikipedia, wie folgende Rechenhilfsmittel funktionieren.
Was können sie und arbeiten sie analog oder digital?
 - a. Rechenstab
 - b. Abakus

2. Rechnerarchitektur nach v. Neumann
 - a. Beschreiben Sie die Arbeitsweise eines derartigen Computers.
 - b. Welche Aufgaben kann ein v. Neumann-Computer mit leerem Hauptspeicher lösen?
 - c. Welche Befehlsarten sind für einen Universalcomputer unbedingt erforderlich?
 - d. Welche Adressierungsarten gibt es?
 - e. Worin bestehen die Grenzen der v. Neumann-Architektur?

3. Einfache Mikrorechner
 - a. Beschreiben Sie grob den Aufbau eines Mikrocomputers.
 - b. Wie bezeichnet man Steuer- und Rechenwerk (in engl. Sprache)?
 - c. Welchen Vorteil bringt die Aufteilung der Adressen in Segment- und Offsetadresse?

4. Kellerspeicher
 - a. Wie greift man auf den Kellerspeicher schreibend bzw. lesend zu?
Was ist dabei zu beachten?
 - b. Ein Hauptprogramm legt einen Registerinhalt als Input-Parameter in den Stack und ruft dann mit „CALL“ ein Unterprogramm auf. Dieses rettet zunächst 5 Register durch Schreiben in den Stack und beginnt dann mit der Bearbeitung. Ein Rückgabewert wird in ein Register geschrieben.
Was muss vor dem „RETURN“-Befehl programmiert werden?
Was kann bei fehlerhafter Benutzung des Stacks passieren?
 - c. Was kann passieren, wenn der Stack zu klein dimensioniert wird?

5. Was versteht man unter Echtzeitverarbeitung?

6. Privilegien

- a. Wie können Anwenderprogramme bei einfachen Computern/Betriebssystemen die Arbeit des Betriebssystems stören?
- b. Was passiert bei einfachen Computern/Betriebssystemen, wenn ein Anwendungsprogramm eine Endlosschleife enthält?
- c. Warum ist die Privilegierung des BS-Kernes sinnvoll?
Was dürfen Anwendungsprogramme in fortgeschrittenen Systemen nicht?
- d. Erfordert die Privilegierung Hardwareunterstützung?
Löst die Privilegierung das Endlosschleifenproblem?

7. Synchrone/asynchrone Arbeit

- a. Welche Vorteile bringt die asynchrone Arbeit von Prozessor und Geräteperipherie?
- b. Welche Anforderungen werden an die Geräte gestellt?
- c. Welchen Nutzen bringt ein Prozessorcache?

8. Unterbrechungsbehandlung

- a. Wozu dienen Unterbrechungen?
- b. Ist zur Realisierung eine Hardwareunterstützung erforderlich?
- c. Ein Programm ruft ein Unterprogramm auf.
Vergleichen Sie die Belegung des Stacks ohne Auftreten eines Interruptes mit der Belegung mit Auftreten einer Unterbrechung.
- d. Kann innerhalb einer Interruptbehandlung ein weiterer Interrupt behandelt werden?
- e. Bei einem Intel-Prozessor haben Maus und Netzwerkkarte die gleiche IRQ-Zuordnung.
Gibt es ein Problem?

9. Treiber

- a. Welche Aufgaben hat ein Treiber?
- b. Wie erfolgt die Bearbeitung eines E/A-Auftrages in einem Treiber?
Diskutieren Sie die Rolle des Unterbrechungsbehandlungssystems.

10. Festplatten

- a. Welchen Nutzen hat ein Festplattencache?
- b. Rekonstruieren Sie das Datenstück 2 in einem RAID4-System mit drei Daten- und einer Paritätsfestplatte.
D1: 101100 D2: D3: 011011 P: 111000

11. Virtueller Hauptspeicher

- a. Welchen Vorteil bietet die gestreute Adressierung?
- b. Ist Hardwareunterstützung erforderlich?
- c. Wie groß ist eine Seite beim Intel-Pentium?
- d. Warum erfolgt die Berechnung der Rahmenadresse mehrstufig?

12. Parallele Systeme

- a. Was leistet ein MIMD-System nach Flynn?
- b. Was ist die Aufgabe einer Prozessorpipeline?
- c. Warum haben moderne Prozessoren mehrere Kerne?
- d. Ist die klassische Programmierungstechnik auf hochparallele Systeme übertragbar?

Übung 3.1 (Programmierungstechnik)

1. Einfache Datentypen

- a. Was spielt sich ab im Computer bei Abarbeitung der Anweisung „ $i = j$;“ ?
- b. Welches Problem tritt auf, wenn Sie eine Anweisung „ $i = x$;“ schreiben und i vom Typ „integer“ und x vom Typ „real“ ist?
- c. Warum muss für Daten vom Typ „integer“ oder „char“ keine Typdefinition erfolgen?
- d. Warum müssen o.g. Daten vor Verwendung deklariert werden?
- e. Welches Problem kann auftreten, wenn sie eine Integervariable als Laufvariable (Zähler) benutzen?

2. Felder

- a. Wieviel Speicherplatz belegt ein Feld mit 10 Integer-Komponenten (int jeweils 4 byte)?
- b. Wie kann ein Programmierer den Fall behandeln, dass zum Programmierzeitpunkt die Feldgröße noch nicht bekannt ist, sondern sich erst bei der Laufzeit ergibt?

3. Strukturen

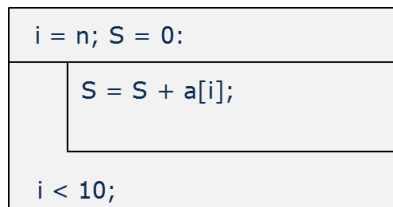
- a. Wozu dient die Typdefinition von Strukturen?
- b. Wozu dient die Variablendeklaration?

4. Zeiger

- a. Was passiert, wenn Sie Inhalte von Zeigervariablen subtrahieren?
 $P3 = \&Z1 - \&Z2;$
- b. Was passiert, wenn Sie Zeigervariable subtrahieren?
 $P3 = *Z1 - *Z2;$

5. Zyklen

- Ein Algorithmus habe 3 ineinander geschachtelte Zählschleifen, die jeweils 10 mal durchlaufen werden.
Wie viele Zyklen durchläuft die innere Schleife insgesamt ?
- Was passiert bei Abarbeitung der Anweisungen des folgenden Struktogrammes ? Beheben Sie evtl. Fehler!

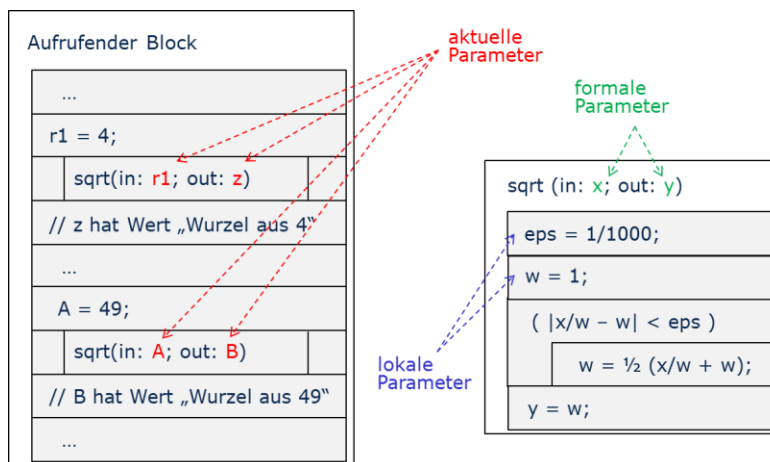


- Wie oft werden folgende Schleifen (Javascript-Notation) durchlaufen?

```
for (i=0; i < 10; i++;) { ... }  
for (i=0; i > 10; i++;) { ... }  
for (i=0; i == 10; i++;) { ... }
```

6. Unterprogramme

- Diskutieren Sie Gemeinsamkeiten und Unterschiede aktueller, formale und lokaler Parameter.
- Was sind globale Parameter?



7. Quadratwurzel

- Wie gut ist (bei obigem Algorithmus) nach 3 Iterationen der Näherungswert für die Berechnung von $\sqrt{9}$?
- Was könnte passieren, wenn im obigen Struktogramm die Bedingung lauten würde:

$$((x/w - w) < \text{eps})$$

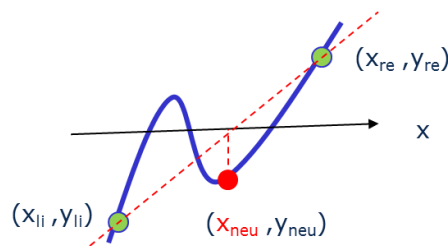
8. Bubble Sort

Wieviele Vertauschungsvorgänge erfolgen bei optimiertem „bubble sort“ für ein Feld von 10 Zahlen, wenn dieses nach Größe sortiert werden soll und

- dieses schon sortiert ist.
- dieses genau gegenläufig sortiert ist.

9. Regula Falsi

- Im Kap. 3.2 auf Folie 17 wird die Anwendung der Nullstellenbestimmungsmethode „regula falsi“ diskutiert. Welche der 3 Nullstellen wird gefunden?



- Was passiert, wenn die Voraussetzungen von „regula falsi“ nicht erfüllt sind (1 Startpunkt unter der x-Achse, 1 Startpunkt darüber)?
- Wie kann die in b) festgestellten Probleme programmtechnisch abgefangen werden?

10. Diskussion der Programmierübungen

Übung 3.2 (Programmierübungen)

0. Vorbereitung

Laden Sie das Archiv „prog-ueb-material.zip“ von der Lehrveranstaltungsseite in einen Testordner und entpacken es dort.

Folgende Dateien liegen dann vor:

- 2 Hilfs-Dateien, die Sie transparent nutzen können und für die keine Bearbeitung erforderlich ist.
 - vkinf-prog-ueb.htm
 - script.css

und

- eine provisorische Javascript-Datei für die nachfolgenden Programmierübungen
 - vki.js

Die Programmieraufgaben beinhalten ausschließlich Übungen zu algorithmischen Grundstrukturen.

Als Programmiersprache wird Javascript wegen der einfachen Handhabung genutzt. Es ist keine umfassende Einführung in Javascript vorgesehen, alle für die Erfüllung der Aufgaben „unwichtigen“ Codezeilen sind in der Datei „vki.js“ bereits enthalten.

Weiterführende Informationen zu Javascript können Interessenten in folgenden Quellen finden:

<http://js-tut.aardon.de/js-tut/index.html#v3>

<http://de.selfhtml.org/javascript/>

Für alle Aufgaben ist zunächst eine Lösungskonzeption zu erarbeiten (PAP oder Struktogramm oder Pseudocode).

Danach sind in der Javascript-Datei „vki.js“ die freigelassenen Codezeilen auszufüllen.

Getestet wird durch Aufruf der HTML-Datei „vkinf-prog-ueb.htm“ in einem aktuellen Webbrowser.

Ablauf:

1. In der Testseite „vkinf-prog-ueb.htm“ ist für jede Aufgabe_i ein Formular mit einer Tabelle enthalten.
Die Input-Parameter werden zunächst in das Formular eingetragen. (Beachten Sie, dass für evtl. Abtrennen von Dezimalstellen nicht das Kommazeichen sondern ein Punktzeichen verwendet werden muss).
Danach werden durch Betätigen der Taste „Calculate“ die Input-Daten des Formulars einer Javascript-Funktion „Aufgabe_i“ in der Datei „vki.js“ übergeben.
2. Alle Funktionen „Aufgabe_i“ besitzen
 - einen vorgefertigten Eingangsteil, in dem die Formulardaten in verarbeitbare numerische Variable umgewandelt werden,
 - einen Verarbeitungsteil, der Gegenstand der Übungsaufgabe ist
(Namen der Variablen sind nicht frei wählbar!)
 - und einen abschließenden Teil für die Ausgabe der Ergebnisdaten in das Aufgabenformular.
3. Nach Rückkehr der Funktion „Aufgabe_i“ in das Webformular kann der Aufruf mit weiteren Testdaten wiederholt werden.

1. Programmieren Sie im Verarbeitungsteil der Funktion „Aufgabe_1“
Anweisungen zur:

- Berechnung der Summe S und des arithmetischen Mittelwertes M aus den 10 Zahlen A, B, \dots, J
- Testen Sie den Algorithmus

2. Programmieren Sie im Verarbeitungsteil der Funktion „Aufgabe_2“
Anweisungen zur:

- Berechnung der Summe S und des arithmetischen Mittelwertes M aus den 10 Feldelementen $a[0], a[1], \dots, a[9]$

Verwenden Sie dafür eine Zählschleife „for“

- Testen Sie den Algorithmus

3. Programmieren Sie im Verarbeitungsteil der Funktion „Aufgabe_3“
Anweisungen zur:

- Berechnung der Summe S und des arithmetischen Mittelwertes M aus den 10 Feldelementen $a[0], a[1], \dots, a[9]$

Verwenden Sie dafür eine kopfgesteuerte Schleife „while“ und brechen Sie die Summation beim ersten unbelegten Feldelement ($a[i] = 0$) ab.

- Testen Sie den Algorithmus

4. Programmieren Sie im Verarbeitungsteil der Funktion „Aufgabe_4“ Anweisungen zur:

- Sortierung der Feldelemente $a[0], \dots, a[9]$ nach abfallender Größe und Ausgabe in einem Feld mit den Elementen $b[0], \dots, b[9]$
- Nutzen Sie den „bubble sort“-Algorithmus für die Sortierung.
- Testen Sie den Algorithmus

5. Verbessern Sie in Aufgabe_5 das Programm von Aufgabe_4, indem Sie den Algorithmus vorzeitig beenden, wenn bereits eine richtig sortierte Reihenfolge der Elemente vorliegt.

6. Programmieren Sie im Verarbeitungsteil der Funktion „Aufgabe_6“ Anweisungen zur:

- Berechnung des Funktionswertes eines Polynoms $p(x)$

Verwenden Sie den Algorithmus „Horner Schema“.

Das Funktionsargument wird durch die Variable x charakterisiert. Die Feldelemente $a[9], \dots, a[0]$ charakterisieren die Koeffizienten des Polynoms (mit max. Grad 9).

Die Koeffizienten des sich durch die Polynomdivision des Horner Schemas ergebenden Polynoms und der Divisionsrest sind in den Feldelmenten $b[9], \dots, b[0]$ abzuspeichern.

Der Funktionswert des Polynoms ist in p zu speichern.

- Testen Sie den Algorithmus für:

$$p = x^2 - 2x + 1 \quad \text{für die Argumente } x = 0, 1, 2$$

und

$$p = x^3 - 2x - 10 \quad \text{für die Argumente } x = 0, 1, 2$$

7. Programmieren Sie im Verarbeitungsteil der Funktion „Aufgabe_6“ Anweisungen zur:

- Berechnung der Nullstellen von Polynomen $p(x)$

Verwenden Sie den Algorithmus „regula falsi“.

Zu Berechnung der Funktionswerte nutzen Sie den Algorithmus von Aufgabe_6 in einer ausgelagerten Funktion „polynom“.

Das Funktionsargument wird durch die Variable x charakterisiert. Linke und rechte Schranke des Betrachtungsintervalles werden durch die Argumente xli und xre charakterisiert.

Der Abbruch der Berechnungen soll erfolgen, wenn die Genauigkeitschranke eps für die Nullstelle erreicht wird.

Die Zahlen a_9, \dots, a_0 charakterisieren die Koeffizienten des Polynoms (mit max. Grad 9).

Ausgegeben werden soll eine Zeichenkette „error“ in die Variable $kode$, falls regula falsi nicht ausgeführt werden darf.

Im positiven Fall soll die Nullstelle in der Variable x gespeichert werden.

- Testen Sie den Algorithmus für:

$$p = x^2 - 2x + 1 \quad \text{für die Argumente } x = 0, 1, 2$$

und

$$p = x^2 - 4x + 1 \quad \text{für die Argumente } x = 0, 1, 2$$

Übung 4 (Softwaretechnologie)

Aus Zeitgründen werden keine Übungen zu diesem Kapitel durchgeführt.

Der Inhalt des Kapitels

wird ausführlich

in der weiterführenden Lehrveranstaltung „Informatik-II“

behandelt.

Übung 5 (Betriebssysteme)

1. Welche Aufgaben hat der Urlader eines Betriebssystems?
 - a. Wie kommt er in den Hauptspeicher?
 - b. Was leisten komfortable Lader?
2. Routinen
 - a. Nennen Sie ein Beispiel für eine Betriebssystemroutine.
 - b. Wodurch unterscheiden sich residente von transienten Betriebssystemroutinen?
 - c. Welche Vor- und Nachteile haben beide Realisierungsformen?
3. Betriebsmittel
 - a. Was für Betriebsmittel gibt es? Nennen Sie Ordnungskriterien.
 - b. Wie kann die Semaphortechnik zur Verwaltung exklusiver Betriebsmittel genutzt werden?
4. Welche Aufgaben hat eine Betriebssystemanlaufroutine?
5. Beschreiben Sie Gemeinsamkeiten und Unterschiede bei Grafik-Ausgaben auf Bildschirm bzw. auf Drucker.
6. Beschreiben Sie Gemeinsamkeiten und Unterschiede bei Ein- und Ausgaben bei Magnetband und Drucker.
7. Dateien
 - a. Was ist eine Datei?
 - b. Beschreiben Sie zwei Dateizugriffsmethoden und diskutieren Sie die zugehörigen Nutzerfunktionen.
 - c. Welche Aufgaben hat die Datenträgerspeicherorganisation?
 - d. Beschreiben und bewerten Sie das FAT-Dateisystem.
8. Hauptspeicherverwaltung
 - a. Welche Aufgaben hat die Hauptspeicherverwaltung?

- b. Welche Vorteile hat die Buddy-Technik gegenüber der Zuteilung des ersten geeigneten Hauptspeicherbereiches?
- c. Was unterscheidet Seiten und Rahmen bei einem virtuellen Speicherkonzept?
- d. Beschreiben Sie die Seitenwechseltechnik (Paging).
Welche Informationen enthalten Seitentabellen?
Wann muß auf die Tabellen zugegriffen werden?
Wodurch kann Seitenflattern entstehen?

9. Prozessverwaltung

- a. Was charakterisiert einen Betriebssystemprozeß?
- b. Entwerfen Sie ein Szenarium, bei dem ein Prozeß alle Globalzustände durchläuft.
- c. Welche Aufgabe hat ein Prozeßzustandsblock PCB?
- d. Sollten normalerweise E/A-intensive Prozesse eine hohe oder niedrige Priorität bekommen (Begründung)?
- e. Auf welche Weise kann das Betriebssystem einem aktiven Prozeß den Prozessor entziehen?
- f. Was versteht man unter einem kritischen Abschnitt eines Prozesses?
- g. Wie können mehrere Prozesse gemeinsamen Programmcode nutzen?
- h. Was ist eine Prozessverklemmung?
- i. Wie kann eine Prozesskommunikation organisiert werden?

10. Welche Vorteile kann die Stapelverarbeitung bringen?

11. Mehrnutzerbetriebssysteme

- a. Was sind die Aufgaben des Administrators?
- b. Wie wird der Dateischutz bei Unix organisiert?
- c. Nennen Sie drei Eigenschaften, die eine Mehrnutzerbetriebssystem aufweisen muß?

12. Welche Vor- und Nachteile bietet die Client-/Server-Architektur gegenüber der monolithischen Architektur?

13. Portabilität

- a. Auf welche Weise kann Software-Portabilität erreicht werden?
- b. Wie kann die Portabilität von Betriebssystemen erreicht werden.

Übung 6 (Datenbanksysteme)

Entfällt aus Zeitgründen

Übung 7 (Rechnernetze)

1. Welche der OSI-Schichten beschäftigt sich jeweils mit
 - a. Signalübertragung
 - b. Nachrichtenaustausch zwischen heterogenen Systemen
 - c. Wegewahl
 - d. Ende-zu-Ende-Kommunikation
2. Vergleichen Sie OSI- und Internetschichtenarchitektur.
3. Diskutieren Sie Vor- und Nachteile verschiedener Übertragungsmedien.
4. Was versteht man unter strukturierter Verkabelung?
5. Wie wirkt sich ein bandbreitenbegrenzter Kanal auf die Signalübertragung aus?
6. Wodurch ist die erreichbare Übertragungsrate eines Nachrichtenkanales beschränkt?
7. Was versteht man unter Modulation? Wieviel Information trägt eine reine Sinuswelle? Welche Aufgaben hat ein Modem?
8. Beschreiben Sie das Prinzip der xDSL-Technik.
9. Wie füegt sich die IEEE802-Architektur in das OSI-Referenzmodell ein?
10. Stellen Sie die verschiedenen Adressierungsebenen des Internet vor.
11. Wie kann ein Intranet an das Internet angekoppelt werden?
12. Welche Aufgaben hat das Internet-DNS?
13. Beschreiben Sie das Prinzip der Web Services.