

Praktikum Kommunikationsinformatik

Projekte Sommersemester 2014

1	Einleitung	2
1.1	Ziel	2
1.2	Inhalt des Praktikums	2
1.3	Ablauf.....	2
1.4	Projektmanagement	3
1.5	Dokumentation.....	4
1.6	Präsentation.....	4
2	Themen.....	4
2.1	Thema 1 (SORT-C): Visualisierung von Sortialgorithmen (C/ C++)	5
2.2	Thema 2 (SORT-J): Visualisierung von Sortialgorithmen (Java)	5
2.3	Thema 3 (GRAPH): Visualisierung von Suchalgorithmen in Bäumen bzw. Graphensuchalgorithmen.....	5
2.4	Thema 4 (4WIN): Vier gewinnt	6
3	Themenwahl	6

1 Einleitung

1.1 Ziel

Der Student erwirbt **praktische Erfahrungen** anhand **informatikspezifischer** und **telekommunikationsspezifischer** Aufgabenstellungen innerhalb von **Projektteams**.

1.2 Inhalt des Praktikums

Entsprechend der Modulbeschreibung:

<http://www-crypto.htw-saarland.de/cgi-bin/moduldb-c?bkeys=ki&ckeys=pk6&lang=de>

- Einführung in das Projektmanagement
- Bearbeitung informatikspezifischer und telekommunikationsspezifischer Themen in Kleingruppen.

Themen aus der Informatik (Weber, Bohr):

- Softwareentwicklung
- Systementwicklung
- TCP/IP
- Algorithmen
- IT-Sicherheit

Themen aus der Telekommunikation (Wieker, Krauss):

- Protokollanalyse
- No.7-Schnittstelle
- TMN/SDH

1.3 Ablauf

Das Praktikum umfasst im Wesentlichen

- Implementierung
- Dokumentation
- Vorbereitung der Präsentation
- Projektkoordination

Den Studierenden steht für den Informatik-Teil der Aufgabe das KI-Labor zur Verfügung.

Das Projektthema ist sehr selbständig im Team zu bearbeiten. Dabei sind grundlegende Gegebenheiten des Projektmanagements anzuwenden.

Präsenz-Termine:

- Vorstellung der Themen am 09.04.2014, 10:00 Uhr, Saal 5104
- Kurzeinführung Projektmanagement
- Kick-Off-Meeting am 25.04.2014
- Vorlage Projektplan am 09.05.2014
- Wöchentliches Status-Meeting: freitags, 1. Stunde
- lt. Stundenplan alle Fächer mit 50% mehr Stundenanteil, wegen vorgezogenem Vorlesungsende am 27.06.2014
- Fertigstellung und Abgabe der Unterlagen am 23.06.2014, 12:00 Uhr mittags

- Präsentation am 27.06.2014

Zeit-Rahmen:

Die Präsenzzeit dieses Moduls umfasst bei 15 Semesterwochen 120 Stunden. Der Gesamtumfang des Moduls beträgt bei 9 Creditpoints 270 Stunden. Daher stehen für die Vor- und Nachbereitung der Veranstaltung zusammen mit der Prüfungsvorbereitung **150 Stunden** zur Verfügung (Informatik-Teil).

Die Aufgabe soll Ende **KW25** abgeschlossen sein, d.h. die Studierenden haben **gut 8 Wochen** Zeit, ihr Praktikumsthema zu bearbeiten. Die verbleibenden 1 – 2 Wochen sind für die Präsentation der Arbeiten vorgesehen. Für das Ende des KI-Praktikums - inkl. Präsentationstermin - ist der **27.06.2014** festgesetzt. Die Projektunterlagen und Datenträger müssen bereits am **23.06.2014** abgegeben werden.

Anmerkung für die Projektplanung:

Zwar dauert die Vorlesungszeit 15 Wochen. Aufgrund der anschließenden Bachelor-Arbeiten ist der Endtermin für das KI-Praktikum vorgezogen. Der Bearbeitungsaufwand für das KI-Praktikum verringert sich dadurch nicht.

Notenermittlung:

Die Endnote für das KI-Praktikum setzt sich zusammen aus jeweils dem Praktikumsergebnis bei Harald Krauss und Damian Weber bzw. Marion Bohr.

Die Gewichtung der Note für das Praktikum bei Damian Weber und Marion Bohr beträgt in etwa:

- Implementierung 50%
- Dokumentation 25%
- Präsentation 25%

Abzugebende Unterlagen:

- (Ausgedruckte) Ausarbeitung, lauffähiges Programm inkl. Quelldateien auf CD, USB-Stick o.ä.
- Präsentation per eMail (im Anschluß an den Vortrag)

1.4 Projektmanagement

Um einigermaßen gleiche Voraussetzungen für alle Studierenden im KI-Praktikum zu schaffen, wird zu Beginn des Praktikums (im Kick-Off-Meeting) eine **Kurzeinführung ins Projektmanagement** gegeben, in dem die wichtigsten Methoden und Instrumentarien vorgestellt werden. Ein einheitliches Excel-Dokument zur Erfassung der Aufwände und des Projektfortschritts (Projektstatusübersicht) wird den Projektleitern zur Verfügung gestellt. Die Projektleiter werden im Kick-Off-Meeting festgelegt.

Tätigkeiten und Projektstatus sind **wöchentlich** über den jeweiligen **Projektleiter** an den Auftraggeber zurückzumelden.

Das setzt voraus, daß sich die Gruppe zu Beginn des Praktikums eine klares Bild von der Aufgabenstellung macht und einen Zeitplan, genauer: **Projektplan**, aufstellt. Anschließend ist die Arbeit auf das Team zu verteilen.

Pro Woche per eMail einzureichen:

- Tätigkeitsbericht: Stichworte zu Tätigkeit pro Mitarbeiter in der abgelaufenen Woche
- Pro Woche hat jedes Teammitglied ein definiertes Soll zu erfüllen. Am Ende der Woche legt jeder Studierende seinem Projektleiter einen Projektstatusbericht vor, d.h. jeder berichtet kurz, was er gemacht hat, wie viel Zeitaufwand er hierfür gebraucht hat und wie viel Zeit er bis zu Fertigstellung seines Anteils am Gesamt-Projekt schätzungsweise noch benötigt. Der Projektleiter fasst dies in der Statusübersicht zusammen:

Studiengang KJPI
Projektarbeit/Praktikum I

Hochschule für Technik und Wirtschaft des Saarlandes
Fakultät für Ingenieurwissenschaften

Lfd.-Nr.	Name, Vorname	Punkte-Äquivalent	Leistungs-Anteil [%]	Summe WS [h]	WW	WW	WW	WW	WW	WW	WW	WW	WW	WW	WW	
					18.04.14	25.04.14	02.05.14	09.05.14	16.05.14	23.05.14	30.05.14	06.06.14	13.06.14	20.06.14	27.06.14	04.07.14
					WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	WS[h] FG[%]	
		20		150	1	2	3	4	5	6	7	8	9	10	11	12
01	Studierender Eins	0,00	0,00%	0												
02	Studierender Zwei	0,00	0,00%	0												
03	Studierender Drei	0,00	0,00%	0												
04	Studierender Vier	0,00	0,00%	0												
05	Studierender Fünf	0,00	0,00%	0												
06	Studierender Sechs	0,00	0,00%	0												
07	Studierender Sieben	0,00	0,00%	0												
08	Studierender Acht	0,00	0,00%	0												
09	Studierender Neun	0,00	0,00%	0												
10	Studierender Zehn	0,00	0,00%	0												

Projekt <Projektname>
<Projektsponsor>

SoSe | WiSe <Jahr>

09.04.2014

Abbildung 1: Projektstatus

1.5 Dokumentation

Die Arbeit soll dokumentiert werden:

- den Projektplan mit den Soll- und Ist-Zielen (Gantt-Diagramm)
- die Tätigkeiten und den Projektstatus

Die Ausarbeitung Ihres Projekts soll insbesondere folgende Punkte beschreiben:

- die Hardware (Betriebssystem, Voraussetzungen, ...)
- das Programm (Aufbau, Struktur, Funktion, ...)
- die Schnittstelle(n)
- eine Benutzerführung
- eine evtl. verwendete Datenbank (Datenbankmodell)
- die evtl. notwendige Konstruktion (bei Robotern Bauplan per Lego-Tool, Fotos, Filmsequenz, ...)

1.6 Präsentation

Die Teams stellen ihr Projekt in Form einer Präsentation vor. Als grobe Richtschnur für die Vortragszeit pro Team werden 20 Minuten (max. 30 min, inkl. Vorführung) angesetzt.

Die Präsentation umfaßt **Folienvortrag** und **Vorführung**.

Der Aufwand dafür gehört zu den o.g. 150 h pro Person.

2 Themen

Nachfolgend ein Überblick über die angebotenen Themen mit einer **groben Beschreibung des Aufgabenszenarios**. Detailinformationen folgen bei der Gruppenarbeit.

Alle Aufgaben sind modular aufgebaut. Machen Sie sich zunächst mit der einzusetzenden Hardware und Software sowie der Aufgabenstellung vertraut und beschaffen Sie die nötigen Informationen und Materialien (Ressourcen). Zerlegen Sie die Aufgabenstellung in Teilprobleme, die sie nacheinander lösen und zu einer kompletten, funktionierenden Lösung zusammenfügen.

2.1 Thema 1 (SORT-C): Visualisierung von Sortieralgorithmen (C/C++)¹

Hintergrund

Spätestens ab dem ersten Semester Informatik haben Sie sich mit einer Reihe von Sortieralgorithmen beschäftigt, die Ihnen immer wieder als grundlegendes Handwerkzeug nahe gelegt werden. Diese Verfahren unterscheiden sich zwar durch charakteristische Vorgehensweisen, lassen sich aber oft erst durch das bildhafte Nachspielen verstehen und einprägen. Umgesetzt werden sollen:

1. Heap-Sort (nicht rekursiv)
2. Insertion-Sort
3. Selection-Sort
4. Bubble-Sort
5. Bucket-Sort
6. Merge-Sort
7. Quick-Sort

Daneben gibt es noch eine Vielzahl weiterer Sortieralgorithmen.

Aufgabe

Implementieren Sie o.g. Algorithmen. Erstellen Sie eine Software zur Visualisierung von Sortieralgorithmen, die das jeweilige Verfahren anschaulich darstellt.

Rekursive Lösungen für Sortieralgorithmen brauchen hier zunächst nicht berücksichtigt zu werden.

Programmierung

Die Realisierung soll in der Programmiersprache C/C++ erfolgen. Für die Gestaltung der grafischen Oberfläche kann z.B. QT oder GTK eingesetzt werden.

2.2 Thema 2 (SORT-J): Visualisierung von Sortieralgorithmen (Java)

Wie Thema 1; allerdings soll die Realisierung in der Programmiersprache Java erfolgen.

2.3 Thema 3 (GRAPH): Visualisierung von Suchalgorithmen in Bäumen bzw. Graphensuchalgorithmen

Hintergrund

In Informatik 2 haben Sie typische Graphensuchalgorithmen kennen und anwenden gelernt, z.B.:

1. Breitensuche
2. Tiefensuche
3. Dijkstra-Algorithmus
4. Bellman-Ford-Algorithmus
5. Floyd-Warshall-Algorithmus

Aufgabe

Erstellen Sie eine Software zur Visualisierung von Suchalgorithmen, die die verwendeten Verfahren anschaulich darstellt. Die Suchalgorithmen müssen ebenfalls implementiert werden.

¹ exponatgeeignet

Programmierung

Die Realisierung soll vorzugsweise in der Programmiersprache C/C++ erfolgen. Für die Gestaltung der grafischen Oberfläche kann z.B. QT oder GTK eingesetzt werden.

2.4 Thema 4 (4WIN): Vier gewinnt²

Hintergrund

Das klassische Spiel "Vier gewinnt" wird auf einen 7 x 7 Spielfeld von zwei Spielern gespielt. Jeder Spieler verwendet Spielsteine in einer ihm zugeordneten einheitlichen Farbe. Jeder versucht, dabei vier seiner Steine fortlaufend nebeneinander auf dem Spielfeld anzuordnen; und zwar entweder waagrecht, senkrecht oder diagonal.

Aufgabe

Das Spiel soll als Programm mit einer grafischen Darstellung des Spielfeldes umgesetzt werden. Folgende Varianten sollen berücksichtigt werden:

- Spieler gegen Spieler
- Spieler gegen Computer
- Erkennung der Gewinnbedingung für beide Spieler
- Intelligenz des Computers muss mindestens „zufällig“ sein

Das Spiel soll internetfähig realisiert werden, so dass zwei Spieler an jeweils einem Rechner gegeneinander spielen. Ein dritter, nur als Zuschauer, soll sich ebenfalls anmelden und das Spiel am Bildschirm verfolgen können.

Zusatzanforderungen:

- a) Der Algorithmus soll auch variable Spielfeldgrößen von 4 x 4, 5 x 5, ..., 11 x 11 korrekt verarbeiten können. Ist 7 x 7 wirklich die ideale Spielfeldgröße? Warum?
- b) Was wäre die ideale Größe des Spielfeldes für "3 gewinnt" bzw. "5 gewinnt"? Warum?

Programmierung

vorzugsweise C/C++, graphische Oberfläche, internetfähige Anwendung für (mindestens) drei Beteiligte, d.h. zwei Spieler und 1 - n Zuschauer

3 Themenwahl

- Mail mit **3 priorisierten** Wunschthemen und Team an marion.bohr@htwsaar.de, damian.weber@htwsaar.de
 - Prioritäten (1-3)
 - Team zu 2 - 3 Personen (s. Aufgabenstellung)
- Einsendeschluß: **11.04.2014, 24:00 Uhr**

² exponatgeeignet