

MODULBESCHREIBUNG

Informations- und Kommunikationssysteme II

Kurzzeichen:	M_luK_II
Code:	412
Durchführungszeitraum:	nicht durchgeführt
ECTS-Punkte:	12
Arbeitsaufwand:	360h
Lernziele:	<p>Die Studierenden</p> <ul style="list-style-type: none"> • können die Komplexität von Algorithmen einschätzen, rekursive Datenstrukturen einsetzen und verstehen die formale Beschreibung von Programmiersprachen mittels Grammatiken. • verstehen die Konzepte von relationale Datenbanken, können relationale Datenbanken methodisch korrekt entwerfen und implementieren. • können HTML5, JavaScript und PHP für Webanwendungen einsetzen. • verstehen die Konzepte des Routing und können Router konfigurieren. • können die Methoden des Softwareengineerings an Projekten praktisch anwenden. • können Unix-basierte Systeme aufsetzen und konfigurieren.
Verantwortliche Person:	Martin Studer
Telefon/Email:	++41 (0)81 2862443/martin.studer@ntb.ch
Standort (angeboten):	Chur
Fachbereiche:	Informatik, Kommunikation
Empfohlene Module:	-
Vorausgesetzte Module:	Informations- und Kommunikationssysteme I
Zusätzlich vorausgesetzte Kenntnisse:	keine
Anschlussmodule:	Informations- und Kommunikationssysteme III

ECTS-Punkte pro Kategorie

Kategorie:	Systemtechnik BB (Standard 05) Profilmodule / 12 Punkte Informations- und Kommunikationssysteme (Standard 05) Systemtechnik VZ (Standard 05) Profilmodule / 12 Punkte Informations- und Kommunikationssysteme (Standard 05)
------------	--

Modulbewertung

Bewertungsart:	Note von 1 - 6
----------------	----------------

Leistungsbewertung

Abgesetzte Modulschlussprüfung:	Prüfung nach spezieller Definition
Bemerkungen zur Prüfung:	Am Ende des Semesters findet eine abgesetzte Modulschlussprüfung in fünf Teilen statt. Die Kurse Algorithmen& Datenstrukturen, Datenbanksysteme, HTML5, JavaScript, PHP, Routing Concepts & Protocols sowie Unix bilden je einen Teil der abgesetzten Modulschlussprüfung.
Während des Semesters:	Während der Unterrichtsphase wird im Kurs Semesterprojekt Software Engineering ein Softwareentwicklungsprojekt bewertet. In den drei Kursen Datenbanksysteme, HTML5, JavaScript, PHP sowie Unix wird je ein Prüfung geschrieben. Im Kurs Routing Concepts & Protocols wird eine Prüfung geschrieben.
Bewertungsart:	Note von 1 - 6
Gewichtung:	Während der Unterrichtsphase wird im Kurs Semesterprojekt Software Engineering ein Softwareentwicklungsprojekt (Gewicht 16.666%) bewertet. In den drei Kursen Datenbanksysteme, HTML5, JavaScript, PHP sowie Unix wird je ein Prüfung (Gewicht je 6.667%) geschrieben. Im Kurs Routing Concepts & Protocols wird eine Prüfung (Gewicht 10%) geschrieben. Am Ende des Semesters findet eine abgesetzte Modulschlussprüfung in fünf Teilen statt. Die Kurse Algorithmen& Datenstrukturen (Gewicht 8.333%), Datenbanksysteme (Gewicht 10%), HTML5, JavaScript, PHP (Gewicht 10%), Routing Concepts & Protocols (Gewicht 15%) sowie Unix (Gewicht 10%) bilden je einen Teil der abgesetzten Modulschlussprüfung.
Bemerkungen:	

Kurse in diesem Modul

Algorithmen & Datenstrukturen

Kürzel:	luK_II_A
Code:	41205
Arbeitsaufwand:	30h
Semester:	1
Lernziele:	<p>Die Studierenden</p> <ul style="list-style-type: none"> • können die Laufzeitkomplexität eines Algorithmus in O-Notation abschätzen. • können rekursive Methoden und rekursive Datenstrukturen in eigenen Programmen einsetzen. • verstehen die formale Beschreibung von Programmiersprachen mittels Grammatiken und können die syntaktische Korrektheit von Quelltext überprüfen.
Lerninhalt:	<ul style="list-style-type: none"> • Algorithmen & Komplexität • Rekursion und rekursive Datenstrukturen • Formale Sprachen zur Beschreibung von Programmiersprachen (z.B. EBNF)
Ansprechperson:	Martin Studer
Telefon/EMail:	++41 (0)81 2862443/martin.studer@ntb.ch
Fachbereiche:	Informatik, Kommunikation
Unterrichtssprache:	Deutsch
Leistungsnachweis:	Es findet zusammen mit den Kursen Datenbanksysteme, HTML5, JavaScript, PHP, Routing Concepts & Protocols sowie Unix eine abgesetzte Modulschlussprüfung statt.
Lehr- und Lernmethoden:	Lehrgespräch, Selbststudium, Übungen, Rechnerübungen
Bibliographie:	<p>Diverse Unterlagen (Internet, Bücher).</p> <p>Ergänzende Literaturangaben:</p> <ul style="list-style-type: none"> • Robert Sedgewick: Algorithms in Java. Addison-Wesley, 2002. • Thomas H. Cormen, et. al.: Introduction To Algorithms. MIT Press, 2001.
Kursart:	Blockkurs mit 1 Lektionen pro Woche
Bemerkungen:	Der Kurs findet jede zweite Woche mit 2 Lektionen statt.

Datenbanksysteme

Kürzel:	luK_II_D
Code:	41201
Arbeitsaufwand:	60h
Semester:	1
Lernziele:	<p>Die Studierenden</p> <ul style="list-style-type: none"> • verstehen die Konzepte von relationalen Datenbanken. • sind fähig, Datenbanken methodisch korrekt zu entwerfen und zu implementieren. • können Sachverhalte mit Entity-Relationship- (ER-) Diagrammen modellieren. • können ER-Diagramme in die entsprechenden Tabellen umsetzen und die Normalisierung erklären und anwenden. • können SQL (Structured Query Language) zur Datendefinition und Datenmanipulation anwenden. • können Lösungen für Datenintegrität, Datensicherung und Datenschutz entwickeln. • kennen die Grundlagen von objektorientierten und objektrelationalen Datenbanken.
Lerninhalt:	<ul style="list-style-type: none"> • Grundlagen und Aufbau von Datenbanken • Entwurfsmethoden für Datenbanken • Relationenmodell und Normalisierung • Datendefinition und Datenmanipulation mit SQL • Physische Datenorganisation/Systemarchitektur • Datenintegrität, Datenkonsistenz, Datensicherung, Datenschutz • Postrelationale Datenbanken (z.B. verteilte, temporale Datenbanken) • Objektorientierte und objektrelationale Datenbanken • Embedded SQL (SQLJ) • Java Database Connectivity (JDBC)
Ansprechperson:	Martin Studer
Telefon/EMail:	++41 (0)81 2862443/martin.studer@ntb.ch
Fachbereiche:	Informatik, Kommunikation
Unterrichtssprache:	Deutsch
Leistungsnachweis:	Während der Unterrichtsphase wird eine Prüfung geschrieben. Zusätzlich findet eine abgesetzte Modulschlussprüfung zusammen mit den Kursen Algorithmen & Datenstrukturen, HTML5, JavaScript, PHP, Routing Concepts & Protocols sowie Unix statt.
Lehr- und Lernmethoden:	Lehrgespräch, Selbststudium, Übungen, Rechnerübungen
Bibliographie:	<p>Literaturangaben:</p> <ul style="list-style-type: none"> • Heide Faeskorn-Woyke, Birgit Bertelsmeier, Petra Riemer, Elena Bauer: <i>Datenbanksysteme</i>. Pearson Studium, 2007.

Kursart: Klassenunterricht mit 2 Lektionen pro Woche

HTML5, JavaScript, PHP

Kürzel: IuK_II_H

Code: 41202

Arbeitsaufwand: 60h

Semester: 1

Lernziele: Die Studierenden

- können HTML5 in Webanwendungen einsetzen.
- können Programme in der Programmiersprache JavaScript entwickeln.
- verstehen die grundlegenden Konzepte der Programmiersprache JavaScript.
- können JavaScript in Webanwendungen einsetzen.
- können Programme in der Programmiersprache PHP entwickeln.
- können PHP in Webanwendungen einsetzen.

Lerninhalt: HTML5:

- Neue/geänderte HTML-Elemente (canvas, input, usw.)
- Neue/geänderte DOM-Schnittstelle (History, Drag & Drop, Offline, usw.)

JavaScript:

- Grundlegende Konzepte der Sprache JavaScript (Objekte und Arrays, Funktionen, Funktionale Programmierung, Objekt-Orientierung und Modularisierung).
- Nutzung von JavaScript in Webanwendungen (Manipulation des DOM-Baumes, Event-Handling).

PHP:

- Grundlegende Konzepte der Sprache PHP
- Serverseitige Nutzung von PHP in Webanwendungen

Ansprechperson: Martin Studer

Telefon/EMail: ++41 (0)81 2862443/martin.studer@ntb.ch

Fachbereiche: Informatik, Kommunikation

Unterrichtssprache: Deutsch

Leistungsnachweis: Während der Unterrichtsphase wird eine Prüfung geschrieben. Zusätzlich findet eine abgesetzte Modulschlussprüfung zusammen mit den Kursen Algorithmen & Datenstrukturen, Datenbanksysteme, Routing Concepts & Protocols sowie Unix statt.

Lehr- und Lernmethoden: Lehrgespräch, Selbststudium, Übungen, Rechnerübungen

Bibliographie: Literaturangaben:

- Marijn Haverbeke: Eloquent JavaScript - A Modern Introduction to Programming. <http://eloquentjavascript.net/>, 2013.

Kursart: Klassenunterricht mit 2 Lektionen pro Woche

Routing Concepts & Protocols

Kürzel: IuK_II_R

Code: 41206

Arbeitsaufwand: 90h

Semester: 1

Lernziele:	<p>Die Studierenden</p> <ul style="list-style-type: none"> • kennen die Funktion und Komponenten eines Routers (HW, OS, RAM, ROM, Interfaces). • kennen die Struktur, Inhalt und Interpretation der Routing Tabelle. • können Metrik und Equal cost load balancing erklären. • können verschiedene Metriken erläutern. • können schrittweise Funktionen des Router bei der Paketweiterleitung erklären. • können die Rolle von statischen Routen beschreiben (direkt verbunden, default route, manuelle route), statische Routen konfigurieren und deren Korrektheit prüfen. • können serielle Verbindungen verstehen und konfigurieren (DCE, DTE, Clock Rate). • können Informationen über das Netzwerk mit dem CDP Protokoll sammeln. • können dynamisches Routing und die Funktionen eines Routingprotokolls beschreiben. • können verschiedene Routingprotokolle nach Einsatz und Funktionsweise klassifizieren sowie die Begriffe Autonomous System, interior Gateway Protocol, Exterior Gateway Protocol, Distance Vector, Link State, classful, classless, convergence, administrative distance erläutern. • kennen die Charakteristiken des Routingprotokoll RIPv1, können RIPv1 konfigurieren, verifizieren und troubleshooten. • können die Konzepte Variable Length Subnet Masking (VLSM) und Classless Interdomain Routing (CIDR) und deren Vorteile und Unterschiede gegenüber classful Routing erläutern. • können das Routingprotokoll RIPv2 konfigurieren, verifizieren und troubleshooten und kennen die Unterschiede zu RIPv1. • können die Unterschiede im Verhalten des Routingprozesses und der Routingtabelle bei classless und classful Routing erläutern. • verstehen das Routingprotokoll EIGRP und können es konfigurieren, verifizieren und troubleshooten. • verstehen die Funktionsweise und Vorteile des DUAL Algorithmus bei EIGRP. • können die Funktionsweise, Konzepte und Vorteile des Link-state Routing erläutern. • verstehen das Routingprotokoll OSPF und können es konfigurieren, verifizieren und troubleshooten.
Lerninhalt:	<ul style="list-style-type: none"> • Metriken (Bandwidth, cost, delay, hop count, load, reliability) • Paketweiterleitung (Paket prüfen, L2 entfernen, Routing Tabelle prüfen, neuen L2 anhängen, auf exit interface weiterleiten) • Routingprotokolle (RIP, IGRP, EIGRP, OSPF, IS-IS, BGP) • Routingprotokoll RIPv1 (classful, automatic route summarization, network discovery, Timers: update, holddown, invalid, flush; loop prevention: holddown, count to infinity, split horizon, poison reverse, IP TTL)
Ansprechsperson:	Beat Bigger
Telefon/EMail:	++41 (0)81 2863731/beat.bigger@ntb.ch
Fachbereiche:	Informatik, Kommunikation
Unterrichtssprache:	Deutsch
Leistungsnachweis:	Während der Unterrichtsphase wird eine Prüfung geschrieben. Zusätzlich findet eine abgesetzte Modulschlussprüfung zusammen mit den Kursen Algorithmen & Datenstrukturen, Datenbanksysteme, HTML, JavaScript, PHP sowie Unix statt.
Lehr- und Lernmethoden:	Lehrgespräch, Selbststudium, Übungen, Rechnerübungen
Bibliographie:	<p>Literaturangaben:</p> <ul style="list-style-type: none"> • Cisco Networking Academy: CCNA Exploration 2: Routing Concepts & Protocols. Online-Kurs.
Kursart:	Klassenunterricht mit 4 Lektionen pro Woche
Semesterprojekt Software Engineering	
Kürzel:	luK_II_P
Code:	41203
Arbeitsaufwand:	60h
Semester:	1
Lernziele:	<p>Die Studierenden</p> <ul style="list-style-type: none"> • können die im Kurs Software Engineering besprochenen Softwareentwicklungs-Prozesse in einem Beispielprojekt praktisch umsetzen. • können je nach Anforderungen (requirements) über passende Modelle, Verfahren und Werkzeuge entscheiden. • können ein Softwareprojekt im Team bearbeiten. • haben zu allen Phasen eines Softwareprojektes Erfahrungen gesammelt, das Projekt ausführlich dokumentiert und die Lösung präsentiert. • haben gelernt, an Terminen und Vorgaben entlang zu arbeiten.
Lerninhalt:	<ul style="list-style-type: none"> • Methodik der Softwareentwicklung im Team • spezielle Anforderungen an Kommunikationssoftware • Techniken zur Dokumentation des Entwicklungsprozesses
Ansprechsperson:	Dr. Ulrich Hauser
Telefon/EMail:	++41 (0)81 2863997/ulrich.hauser@ntb.ch
Fachbereiche:	Informatik, Kommunikation

Unterrichtssprache:	Deutsch
Leistungsnachweis:	Während der Unterrichtsphase wird ein Software Entwicklungsprojekt bewertet.
Lehr- und Lernmethoden:	Selbstständige Bearbeitung eines Softwareentwicklungsprojektes in Gruppen.
Kursart:	Blockkurs mit 1 Lektionen pro Woche
Bemerkungen:	Der Kurs findet jede zweite Woche mit 2 Lektionen statt.

Unix

Kürzel:	luK_II_U
Code:	41204
Arbeitsaufwand:	60h
Semester:	1
Lernziele:	<p>Die Studierenden</p> <ul style="list-style-type: none"> • können System-Hardware bestimmen und konfigurieren, den Boot-Prozess begleiten und zwischen Runlevels wechseln. • können das Partitions-Schema eines Linux-Systems konzipieren und einen Boot-Manager konfigurieren. • können gemeinsam genutzte Bibliotheken bestimmen und Paketverwaltungssysteme (Debian, RPM, YUM) zur Installation von Paketen nutzen. • können die Kommandozeile nutzen, um mit Dateien und Verzeichnis, Prozessen und Prioritäten umzugehen. • können Streams nutzen, Filter auf Streams anwenden und Streams verbinden, sowie reguläre Ausdrücke anwenden, um effizient mit textuellen Daten umzugehen. • können mit vi einfache Aufgaben durchführen. • kennen den Filesystem Hierarchy Standard (FHS). • können Dateisysteme auf Datenträgern installieren, unterhalten und mounten. • können den Dateizugriff über Dateirechte und –besitzer steuern und Diskquotas für Benutzer verwalten. • können hard und soft links erstellen und verwalten. • können Dienste installieren und verwalten.
Lerninhalt:	<ul style="list-style-type: none"> • System Architecture • Linux Installation and Package Management • GNU and Unix Commands • Devices, Linux Filesystems, Filesystem Hierarchy Standard
Ansprechsperson:	Dr. Ulrich Hauser
Telefon/EMail:	++41 (0)81 2863997/ulrich.hauser@ntb.ch
Fachbereiche:	Informatik, Kommunikation
Unterrichtssprache:	Deutsch
Leistungsnachweis:	Während der Unterrichtsphase wird eine Prüfung geschrieben. Zusätzlich findet eine abgesetzte Modulschlussprüfung zusammen mit den Kursen Algorithmen und Datenstrukturen, Datenbanksysteme, HTML5, JavaScript, PHP, sowie Routing Concepts & Protocols statt.
Lehr- und Lernmethoden:	Lehrgespräch, Selbststudium, Übungen, Rechnerübungen
Bibliographie:	Frei zugängliche Quellen im Web
Kursart:	Klassenunterricht mit 2 Lektionen pro Woche