

Lehrplan

Klassenstufe 11: Technologie

Klassenstufe 12: Technologie/Technische Mathematik

Fachoberschule

Fachbereich Ingenieurwesen

Fachrichtung Technische Informatik

Ministerium für Bildung, Kultur und Wissenschaft

Hohenzollernstraße 60, 66117 Saarbrücken

Postfach 10 24 52, 66024 Saarbrücken

Saarbrücken 2007

Hinweis:

Der Lehrplan ist online verfügbar unter

www.saarland.de/bildungserver.htm

Einleitende Hinweise

Der vorliegende Lehrplan der Fachoberschule für Ingenieurwesen / Fachrichtung Technische Informatik für das Fach Technologie/Technische Mathematik basiert auf der

- Verordnung - Schulordnung - über die Ausbildung an Fachoberschulen im Saarland vom 24. Juni 1986 (Amtsbl. S. 605), vom 13. April 2006 (Amtsbl. S. 605) sowie der
- Verordnung - Prüfungsordnung - über die staatlichen Abschlussprüfungen an den Fachoberschulen im Saarland (APO-FOS) vom 3. Juli 1981 (Amtsbl. S. 455) in der Fassung vom 13. April 2006 (Amtsbl. S. 605 [613]).

Er erfüllt die Vorgaben der aktuellen

- KMK-Rahmenvereinbarung über die Fachoberschule vom 16.12.2004 und beachtet die Standards der
- KMK- Vereinbarung über den Erwerb der Fachhochschulreife in beruflichen Bildungsgängen vom 05.06.1998 in der Fassung vom 09.03.2001.

Die rasant fortschreitende Computertechnologie (insbesondere im Bereich der Hardware) hat in den letzten Jahren die Voraussetzungen dafür geschaffen, dass immer mehr und leistungsfähigere Rechner im Verbund miteinander (Netzwerk) mit immer intuitiveren Betriebssystemen betrieben werden. Diese Tendenz zur verteilten Rechenleistung führt zu leistungsfähigen Rechner-Netzen, die in der Regel inhomogen aufgebaut sind. Auf der einen Seite stehen die Arbeitsplatzrechner (Workstations), die mit den unterschiedlichsten Betriebssystemen betrieben werden, auf der anderen Seite sind die Server zu finden, deren Netzwerk-Betriebssysteme dafür sorgen, dass die Arbeitsstationen auf gemeinsame Ressourcen zugreifen können.

Das Fach *Technologie/Technische Mathematik* innerhalb der *Fachoberschule, Fachbereich Ingenieurwesen, Fachrichtung Technische Informatik* versucht dieser Problematik dadurch gerecht zu werden, dass nicht nur verschiedene Themenbereiche der theoretischen Informatik, der Hard- und der Software behandelt, sondern insbesondere auch Basiskonzepte und -fertigkeiten vermittelt werden, die plattformübergreifend (hier ist Hard- und Software gemeint) angewendet werden können. Dies ist umso notwendiger, da die Innovationszyklen im Hard- und Softwarebereich sich immer mehr verkürzen und die Änderungsgeschwindigkeiten exponentiell steigen.

Die Schule muss auf geeignete Art und Weise auf diese Veränderung der Hard- und Softwarelandschaft reagieren, d.h. die aktuelle Entwicklung muss ständig beobachtet, Auswirkungen auf die Arbeitsbedingungen der Schüler überprüft und - wenn notwendig - Inhalte und Ausstattung korrigiert werden.

Darüber hinaus soll durch den Kontakt mit Open Source Software das Kostenbewusstsein geweckt und zu ökonomischem Handeln angeleitet werden.

Das Fach *Technologie/Technische Mathematik* bildet zwar eine eigenständige Einheit innerhalb der Fächer der Fachoberschule, sollte aber noch mehr als andere Fächer integrativ zum Fach Visuelle Dokumentation und zum Fach Englisch gesehen werden. Viele Inhalte werden von allen drei Fächern tangiert und einige Inhalte können mit denselben Werkzeugen innerhalb der Informatik und der Visuellen Kommunikation vermittelt werden.

Der Lehrplan verfolgt das Ziel, die für ein Studium an einer Fachhochschule notwendige Studierfähigkeit zu vermitteln. Das Durchdringen komplexer Zusammenhänge soll ermöglicht sowie studienrelevante Fähigkeiten und Einstellungen gefördert werden. Ziel des Lehrplans ist somit nicht die ausschließlich fachsystematische Vermittlung von Wissen in einem einzigen Fachbereich, sondern vielmehr ein Prozess des Aneignens von Kompetenzen im Sinne eines umfassenden Verständnisses der Informatik und ihrer Handlungsfelder.

Insbesondere sollen die Schülerinnen und Schüler für den Ingenieurberuf typische Kompetenzen erwerben:

Fachkompetenz

- Aufgaben und Probleme der Informatik auf der Grundlage von Gesetzmäßigkeiten lösen
- grundlegende Arbeits- und Denkweisen der Informatik anwenden
- die Fachsprache und Symbolik beherrschen sowie informatische / informationstechnische Sachverhalte formulieren

Methodenkompetenz

- Methoden für eine folgerichtige Gedankenführung und ein systematisches Vorgehen beherrschen
- Problemlösungsstrategien anwenden, die ingenieurgerechtem Arbeiten entsprechen
- Informationen beschaffen, strukturieren und auswerten
- Projekte im Team durchführen
- Methoden der Qualitätssicherung anwenden
- Dokumentationen anfertigen und Ergebnisse präsentieren

Personal- und Sozialkompetenz

- Selbstständigkeit
- Verantwortlichkeit
- Zielstrebigkeit
- Ausdrucksfähigkeit
- Teamfähigkeit

Insbesondere in der Klassenstufe 12 soll handlungsorientiert unterrichtet werden. In diesem Unterricht wird mindestens ein integratives Projekt durchgeführt. Die Leistungen, die in der Projektarbeit erbracht werden, gehen in die Vornote ein und sind auch als Gegenstand für die mündliche Abschlussprüfung geeignet.

Lernen in der Fachoberschule berücksichtigt konkretes berufliches Handeln sowie dessen gedankliches Nachvollziehen. Bei der Gestaltung handlungsorientierten Unterrichts sind folgende Orientierungspunkte von Bedeutung:

- Handlungen müssen von den Lernenden möglichst selbständig geplant, durchgeführt, überprüft, ggf. korrigiert und anschließend bewertet werden.
- Handlungen sollen ein ganzheitliches Erfassen der beruflichen Wirklichkeit fördern, z.B. technische, sicherheitstechnische, ökonomische, ökologische und soziale Aspekte einbeziehen.¹

¹Vgl. Handreichungen der KMK zur Erstellung von Rahmenlehrplänen, Bonn, 2001

Auf nachstehende formale Vorgaben wird verwiesen:

- Die Lernziele sind mit Blick auf einen stringenten Umfang des Lehrplans als Grobziele formuliert.
- Die didaktischen Schwerpunkte dieser Lernziele heben hervor, worauf es im Unterricht besonders ankommt. Diese didaktischen Schwerpunkte sind: Wissen, Können, Erkennen und Werten.
- Die Zeitrichtwerte sind als vorgeschlagene zeitliche Empfehlungen zu verstehen. Sie sind als Jahresstunden ausgewiesen, um Vergleiche mit Schulformen anderer Bundesländer zu ermöglichen.
- Nicht ausgewiesen sind Stundenanteile für Wiederholungen, Leistungsüberprüfungen, Vertiefungen, Unterrichtsausfall usw.. Die Lehrplankommission hat diese Anteile in den Zeitrichtwerten der Lerngebiete des Lehrplanes berücksichtigt.

Saarbrücken, März 2007

LERNGEBIETSÜBERSICHT

Klassenstufe 11 - Technologie (2 Wochenstunden)		
Lfd. Nr.	Lerngebiet	Zeitrichtwert * Stunden
1	Baugruppen von DV-Anlagen / Hardware	10
2	Aufbau von Betriebssystemen / Systemsoftware	30
3	Internetdienste und -Strukturen (WWW, E-Mail, FTP]	10
4	Grundlagen der objektorientierten Programmieretechnik	30
Summe		80

*Zeitrichtwert i.S. eines Vorschlags

Klassenstufe 12 – Technologie/Technische Mathematik (6 Wochenstunden)		
Lfd. Nr.	Lerngebiet	Zeitrichtwert * Stunden
5	Datenbankentwurf und SQL	25
6	Objektorientierte Programmieretechnik	108
7	Aufbau und Strukturen von Netzwerken und Netzwerkbetriebssystemen	80
8	Projekte	27
Summe		240

* Zeitrichtwert i.S. eines Vorschlags

Hinweise zu den Lerngebieten

zu 1: Im Lerngebiet *Baugruppen von DV-Anlagen / Hardware* sollte exemplarisch auf aktuelle Bausteine und ihre typischen Merkmale im Rechnerbereich eingegangen werden. Detaillierte Betrachtungen haben aufgrund der immer größer werdenden Komplexität (und Integrationsdichte) wenig Sinn.

zu 2: Das Lerngebiet *Aufbau von Betriebssystemen / Systemsoftware* gliedert sich in zwei Hauptteile: Zum Einen werden die grundlegenden Strukturen moderner Betriebssysteme behandelt. Hierdurch soll der Schüler die Fähigkeit erhalten, sein Wissen zu einem späteren Zeitpunkt mit möglichst geringem Aufwand an das hohe Innovationstempo der Betriebssystem-Entwicklungen anpassen zu können. Die Grundlage zur fachlichen Flexibilität wird hierbei geschaffen.

Zum Anderen soll er auch konkrete Installationen, Einstellungen und Anpassungen eines bestimmten Betriebssystems beherrschen, um die typischen Arbeitsabläufe beim Umgang mit Betriebssystemen und Computern kennen zu lernen. Der Schüler wird später in der Praxis häufig mit der konkreten Aufgabe betraut sein, Betriebssysteme einzurichten, anzupassen und zu warten. Hierzu sind u. a. grundlegende Kenntnisse der Strukturen von Datenträgern nötig. Daher wird vor den praktisch orientierten Lerninhalten der Aufbau und die Struktur von Datenträgern behandelt.

Die modernen, grafisch orientierten Benutzeroberflächen der heutigen Betriebssysteme haben den Vorteil einer gewissen Übersichtlichkeit, erreichen jedoch in mancher Beziehung nicht die Flexibilität einer Kommandozeileingabe. Daher werden typische Strukturen von Kommandozeilenbefehlen beispielhaft behandelt.

zu 3: In diesem Lerngebiet werden dem Schüler grundlegende, technische Zusammenhänge von aktuellen Internetdiensten vermittelt und im Zusammenspiel mit Inhalten aus anderen Lernbereichen (z.B. 7) dargeboten.

zu 4: Zur Vermittlung der Inhalte dieses Lerngebietes können unterschiedliche Softwareplattformen genutzt werden: Delphi stellt aufgrund seiner syntaktischen Verwandtschaft zu Pascal und seiner visuell unterstützenden Entwicklungsumgebung eine gute Basis zur effektiven Arbeit unter grafisch orientierten Betriebssystemen dar. Beispiele zum objektorientierten Ansatz müssen nicht an den Haaren herbeigezogen werden, sondern bieten sich durch die Verwendung vordefinierter Komponenten mit der ersten Kontaktaufnahme mit der Entwicklungsumgebung auf natürlichem Wege sofort an. Ein weiterer Vorteil von Delphi besteht in der Effizienz des erzeugten Codes und der Aussicht auf Portierung auf andere Betriebssystem-Plattformen (LINUX)

Neben Delphi bietet sich auch Java mit seiner C++-ähnlichen Syntax an. Vorteil dieser Plattform ist die breite Verfügbarkeit entsprechender Entwicklungsumgebungen und die unmittelbare Verwendbarkeit der erzeugten Programm-Module innerhalb eines Browsers und die mit dem Internet boomende Verbreitung dieser Klasse von Entwicklungsumgebungen.

zu 5: In diesem Lerngebiet soll der Schüler in die Problematik der Datenbankentwicklung und -verwaltung herangeführt werden. Er lernt Datenbankmodelle kennen und er wird in die Einrichtung, Verwaltung und Pflege von Datenbanken sowie deren Abfragemöglichkeiten eingeführt. Die hier vermittelten Inhalte lassen sich fächerübergreifend in Lerngebiet 2 *Erstellen von Internetdokumenten* des

Faches *Visuelle Dokumentation* sowie in Projekten des Lerngebietes 8 praktisch umsetzen.

zu 6: Dieser Lernbereich kann als Erweiterung oder Ergänzung zum Lernbereich 4 gesehen werden. Es ist denkbar, die gleiche Entwicklungsumgebung wie in der Klasse 11 zu verwenden und den Umgang mit dieser weiter zu vertiefen bzw. erst richtig einzusteigen (Delphi/Delphi bzw. Java/Java). Man könnte aber auch die Syntaxfamilie wechseln und nur auf den im Lernbereich 4 erworbenen allgemeinen Kenntnissen aufbauen (Delphi/Java).

zu 7: Die inhaltliche Konzeption des Lerngebietes 7 wird von außergewöhnlichen äußeren Rahmenbedingungen mitbestimmt: Die Innovationszyklen in der Netzwerktechnik sind wie kaum in einem anderen Teilbereich der Datenverarbeitung äußerst schnellen Entwicklungen und Innovationsschüben unterworfen. Insofern ist es kaum möglich, auf die jeweils aktuellsten Entwicklungen in der Netzwerktechnik inhaltlich im Lehrplan konkret einzugehen.

Des Weiteren ist zu bedenken, ob neue technologische Entwicklungen nach kurzer Zeit sich als obsolet herausstellen oder ob sie neue grundlegende Strukturen für zukünftige technische Anwendungen darstellen.

Aufgrund dieser Überlegungen soll einerseits den jeweils aktuellsten Entwicklungen dadurch Rechnung getragen werden, dass die Schüler zum Beispiel durch Projektarbeit auf praktisch orientierte Art und Weise an die aktuellen Themen der Netzwerktechnik herangeführt werden. Dabei kann es im besonderen bei dieser Projektarbeit vorkommen, dass bestimmte Schüler ein eventuell vorhandenes hochspezialisiertes Vorwissen einbringen und damit für die Motivation und den Lernerfolg der ganzen Klasse nützlich sein können. Andererseits sollen (in Anlehnung an die größtenteils dauerhaft bestehende Lehrplaninhalte) Strukturen und Funktionen von Netzwerken, die sich über längere Zeit bewährt haben und die aller Voraussicht nach länger von Bestand sein werden, in den Lehrplan als feste Bestandteile aufgenommen und behandelt werden.

Als Vermittlungsplattform für das Lerngebiet 7 eignet sich LINUX wegen seiner universellen Einsetzbarkeit (insbesondere auch als Serverbetriebssystem), seiner Leistungsfähigkeit, seiner Vollständigkeit (insbesondere in Bezug auf internetbezogene Serverdienste), seiner Transparenz, seiner hervorragenden Dokumentation und nicht zuletzt wegen seiner günstigen Kosten eher besser als die bisher bekannten Microsoft-Betriebssysteme.

zu 8: In diesem Lerngebiet haben die Schüler die Möglichkeit, fachübergreifende Projekte zu realisieren. Hier können die Lernmodule, die in den ersten 7 Lerngebieten genannt werden, in einer konkreten Aufgabenstellung vertieft werden.

Lerngebiet 1: Baugruppen von DV-Anlagen / Hardware Zeitrichtwert*: 10 Stunden

Lernziele	Lerninhalte	Hinweise zum Unterricht
1.1 Grundbegriffe und Wirkungsweise einer DVA kennen	Blockschaltbild einer DV-Anlage EVA-Prinzip Binär-, Oktal- und Hexadezimalzahlensysteme	<ul style="list-style-type: none"> - Einsatzgebiete - Datenfluss - Transport - Speicherung - Verknüpfung
1.2 Aufbau und Funktionsweise eines Rechnersystems kennen	Prozessor Bussysteme Speicher Schnittstelle	<ul style="list-style-type: none"> - Adress-, Daten-, Steuerbus - interne Speicher: RAM, ROM, EPROM, EEPROM - parallele und serielle Übertragungstechnik und Protokolle
1.3 Peripheriegeräte von DV-Anlagen benennen	Blockschaltbild einer DV-Anlage Eingabegeräte Speichergeräte Ausgabegeräte	<ul style="list-style-type: none"> - Tastatur, Maus, Scanner - Festplatte, CD-ROM, DVD, Floppy-Disk, Flash ROM und neue Entwicklungen - Bildschirm, Tastatur, Plotter

*Zeitrichtwert i.S. eines Vorschlags

Lerngebiet 2: Aufbau von Betriebssystemen / Systemsoftware Zeitrichtwert*: 30 Stunden

Lernziele	Lerninhalte	Hinweise zum Unterricht
2.1 Die geschichtlichen Entwicklung von Betriebssystemen kennen	Ursprung von Betriebssystemen Entwicklung der Betriebssysteme	
2.2 Bestandteile und Aufgaben eines Betriebssystems beschreiben	Betriebssysteme als Mittler zwischen Hard- und Software Typische Aufgaben eines Betriebssystems	Organisations-, Dienst- und Übersetzerprogramme
2.3 Grundsätzliche Konzepte von Betriebssystemen erklären	Monolithische Betriebssysteme Client-Server Betriebssysteme Betriebssysteme mit Mikrokern Kombinationen verschiedener Prinzipien	z.B. WINDOWS XX, AIX, HPUX, Solaris, Irix, Linux, OpenSolaris
2.4 Unterschiede zwischen proprietären und offenen Betriebssystemen bewerten	z.B. Microsoft- vs. offene Betriebssysteme	
2.5 Die Dateiverwaltung unter proprietären und offenen Betriebssystemen erklären	Dateisysteme von WINDOWS und Linux und deren Aufgaben	<ul style="list-style-type: none"> - NTFS - FAT - FAT 32 - Reiser - Raid (Level 0-5)
2.6 Den Aufbau von Festplatten und Partitionen beschreiben	Aufteilungselemente einer Festplatte Erstellen von Partitionen Erweiterte Partition und logische Laufwerke Formatieren von Festplatten Bootvorgang, Bootmanager	<ul style="list-style-type: none"> - Spuren, Zylinder, Köpfe, - Oberflächen Sektoren - Cluster - Master-Boot-Record

2.7	Struktur und Funktionen eines einfachen Boot-Betriebssystems kennen	Struktur eines einfachen Betriebssystems Typische Kommandozeilenbefehle Batch-Dateien	
2.8	Struktur und Funktionen eines Multitasking-Betriebssystems kennen	Designziele und Anforderungen an ein modernes Betriebssystem Struktur und Zusammenwirken der Teile des Betriebssystems Typische Struktur und Funktionen verschiedener Dateisysteme Wichtige Kommandozeilenbefehle	Kernelmodus und Usermodus
2.9	Proprietäre und offene Betriebssysteme installieren und handhaben	Vorüberlegungen und Vorbereitungen Partitionierung der Festplatte Wahl des Dateisystems Grundinstallation des Betriebssystems Anbindung an ein bestehendes Netzwerk	- WINDOWS-Installationsassistent - Yast
2.10	Betriebssysteme an besondere Anforderungen anpassen	Hilfesystem Anpassung von Teilen der Systemsteuerung Eingabeaufforderung Geräte und Dienste Notfallmanagement Diagnosewerkzeuge Taskverwaltung Ereignisanzeige, Systemmonitor Festplattenmanager Einrichtung lokaler Drucker Dateiattribute und Berechtigungen Freigaben, Besitzübernahmen, Überwachung Benutzerverwaltung	Anlage neuer Benutzer Benutzergruppen Benutzerrechte und Benutzerberechtigungen

*Zeitrichtwert i.S. eines Vorschlags

Lerngebiet 3: Internetdienste und -strukturen		Zeitrichtwert*: 10 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
3.1 Grundsätzliche Strukturen des Internets kennen	Entstehung des Internets Struktur des Internets	RFC Internetadressen URL
3.2 Anwendungsorientierte Dienste des Internets kennen	Umgang mit Browsern HTTP FTP E-Mail Telnet News	
3.3 Methodisches Vorgehen bei der Nutzung von Internetdiensten beherrschen	Thematische Suche Komplexe Stichwortsuche Geographische Suche Verwendung des FTP Dienstes Einrichtung einer Homepage	
3.4 Sicherheitsaspekte des Internets kennen und entsprechende Maßnahmen anwenden	Bewusstmachung des Themas Sicherheit Sicherheitsproblematische Strukturen des Internets Grundsätzliche Methoden zur Sicherheitserhöhung Beispiele für aktuelle Sicherheitsstrategien Grundsätzliche Methode zur Erhöhung der Sicherheit	Firewall Verschlüsselung, z.B. PGP, GnuPG SSL

*Zeitrichtwert i.S. eines Vorschlags

Lerngebiet 4: Grundlagen der objektorientierten Programmierertechnik		Zeitrichtwert*: 30 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
4.1 Informatische Modelle exemplarisch erstellen und diese in die Schritte der Programmentwicklung einordnen	Vorgang der Programm-erstellung	Problemformulierung, Problemanalyse Konzeption notwendiger Datenstrukturen Entwerfen von Algorithmen Problemlösungsstrategien Programmablaufpläne
4.2 Informatische Modelle mittels ausgewählter Programmierparadigmen in Softwarebausteine umsetzen	Elemente einer visuell unterstützten Entwicklungsumgebung	
4.3 Einfache Programme selbstständig entwickeln Einfache Programme mit Hilfe eines visuellen Softwareentwicklungssystems entwickeln	Aufbau eines Programms Grundlegende Sprach- elemente und ihre Eigen- schaften Standard-Datentypen Objektorientierte Prob- lem-lösung und Ereignis- steuerung	Kommentare, Bezeichner, Symbole, reservierte Wörter, Konstanten, Variablen, Aus- drücke Zeichen, Zeichenketten, Ganzzahlen, Gleitkomma- zahlen, Wahrheitsdatentyp bzw. bool'sche Bewertung von Ausdrücken Referenzen bzw. Zeiger Instanz eines Klassentyps Methoden und Eigenschaften als Klassenkomponenten Kapselung, Vererbung Sichtbarkeit von Klassenkom- ponenten Arten von Methoden, Aktivie- rung und Implementierung von Methoden Klassenoperatoren Behandlung von Botschaften des OS

	<p>Ausdrücke</p> <p>Benutzung visueller Komponenten</p> <p>Kontrollstrukturen</p>	<p>Operatoren und ihre Prioritäten</p> <p>Zeichenketten-Ausdrücke</p> <p>Arithmetische Ausdrücke</p> <p>Logische Ausdrücke</p> <p>Bedingte Programmierverzweigungen</p> <p>Zählwiederholung</p> <p>Abweisende Wiederholstruktur</p> <p>Nichtabweisende Wiederholstruktur</p>
--	--	---

*Zeitrichtwert i.S. eines Vorschlags

Lerngebiet 5: Datenbankentwurf und SQL		Zeitrichtwert*: 25 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
5.1 Einfache Datenbanken entwerfen	<p>Datenmodellierung relationaler Datenbanken</p> <p>Darstellungsformen</p> <p>Transformation des ERM</p> <p>Normalisierung</p>	<p>Entity-Relationship-Modell (ERM)</p> <p>Beziehungstypen (1:1, 1:N, N:M-Beziehung)</p> <p>Primärschlüssel</p> <p>Darstellung in Tabellenform</p> <p>Datensatz, Datenfeld</p> <p>Transformation des ERM in Tabellenform</p> <p>Referentielle Integrität</p> <p>Fremdschlüssel</p> <p>Sinn und Zweck der Normalisierung</p> <p>Erste Normalform</p> <p>Zweite Normalform</p> <p>Dritte Normalform</p>
5.2 Einfache Datenbanken einrichten	<p>Datenbankstruktur anlegen in SQL</p> <p>SQL-Datenmanipulation</p>	<p>SQL-Datentypen</p> <p>CREATE TABLE-Anweisung</p> <p>DROP-TABLE-Anweisung</p> <p>INSERT-Anweisung</p> <p>UPDATE-Anweisung</p> <p>DELETE-Anweisung</p>
5.3 Einfache Datenbanken auswerten	<p>SQL-Datenbankabfragen auf eine Tabelle</p> <p>SQL-Datenbankabfragen auf mehrere Tabellen</p>	<p>SELECT-Anweisung</p> <p>Abfrage-Bedingungen (WHERE-Klausel)</p> <p>Sortieren von Daten</p> <p>Gruppieren von Daten</p> <p>Aggregatfunktionen</p> <p>DISTINCT-Klausel</p> <p>Unterabfragen</p> <p>Inner Joins</p> <p>Outer Joins (Left, Right)</p>
5.4 Einfache Datenbanken pflegen und verwalten	Indizes	<p>Begriff der sequentiellen Datensuche</p> <p>Sinn und Zweck von Indizes</p> <p>Datenfelder mit Indizes versehen</p>

	<p>Transaktionen</p> <p>Mehrfachzugriff</p>	<p>Bedeutung von Transaktionen</p> <p>Praktische Realisierung einer Transaktion</p> <p>Auftretende Phänomene: Dirty Read, Lost Update Unrepeatable Read</p> <p>Isolationslevel</p>
--	---	--

*Zeitrichtwert i.S. eines Vorschlags

Lerngebiet 6:		Objektorientierte Programmier- technik	Zeitrichtwert*: Stunden	108
Lernziele		Lerninhalte	Hinweise zum Unterricht	
6.1	Informatische Modelle exemplarisch erstellen und diese in die Schritte der Programmentwicklung einordnen	Methoden zur Erstellung von Programmen	Problemformulierung, Problemanalyse Konzeption notwendiger Datenstrukturen Entwerfen von Algorithmen Problemlösungsstrategien Top-Down-Methode Programmablaufpläne	
6.2	Informatische Modelle mittels ausgewählter Programmierparadigmen in Softwarebausteine umsetzen	Elemente einer visuell unterstützten Entwicklungsumgebung	Dateihandling Editorbedienung Codeerzeugung und -optimierung Fehlersuche mittels Debugger Komponentenbibliotheken	
6.3	Einfache Programme selbstständig entwickeln Einfache Programme mit Hilfe eines visuellen Softwareentwicklungssystems entwickeln	Grundlagen der visuell unterstützten Programm-entwicklung Grundlegende Sprach- elemente und ihre Eigen- schaften	Kommentare, Bezeichner, Symbole, Reservierte Wörter Konstanten Variablen Ausdrücke	
6.4	Theorie der objekt-orientierten Pro-grammentwicklung kennen und verstehen Praxisbezogene Problemstellungen objektorientiert analysieren, modellieren und gefundene Lösungsalgorithmen adäquat implemen-tieren	Klassen und Objekte	Instanz eines Klassentyps Methoden und Eigenschaften als Klassenkomponenten Kapselung, Vererbung Sichtbarkeit von Klassenkom- ponenten Arten von Methoden, Aktivie- rung und Implementierung von Methoden Konstruktoren und Destrukto- ren Klassenoperatoren Behandlung von Botschaften	

		des OS
	Visuelle Komponenten	Benutzung visueller Komponenten Anpassung visueller Komponenten
	Struktur von Programmen	
	Standarddatentypen und ihre Anwendungsbereiche	Zeichendatentyp(en) Ganzzahldatentypen Gleitkommatentypen Wahrheitsdatentyp bzw. bool'sche Bewertung von Ausdrücken Referenzen, Zeiger
	Ausdrücke	Operatoren und ihre Prioritäten Zeichenketten-Ausdrücke Arithmetische Ausdrücke Logische Ausdrücke
	Kontrollstrukturen	Bedingte Programmverzweigungen Zählwiederholung Abweisende Wiederholstruktur Nichtabweisende Wiederholstruktur
	Fehlerbehandlung	Ein-/Ausgabeprüfung Abfangen von Ereignissen bei der Fehlerbehandlung
	Aufbau und Verwendung von Unterprogrammen	Sinn und Zweck Arten der Parameterübergabe Aufruf von Unterprogrammen Prototyping Lokale und globale Variablen Rekursion
	Aufbau und Verwendung von Bibliotheksmodulen	
	Zeichenketten	Beschreibung und Aufbau von Zeichenketten Zeichenkettenoperationen Zeichenkettenfunktionen Umwandlungsfunktionen Manipulation von Zeichenketten
	Definition von Datentypen	Allgemeine Regeln für Skalare

	pen Mengen Felder Dynamische Daten- strukturen Dateien Benutzung und Anpas- sung von Dialogen Benutzung von Grafik- Komponenten Zeiger bzw. Referenzen	Datentypen Aufzählungstypen Teilbereichstypen Struktur von Feldern Felddeklaration Zugriff auf Feldelemente Verarbeitung von Feldern Mehrdimensionale Felder Unterschiedliche Dateiarten Datei-Zugriffsfunktionen
--	--	--

*Zeitrichtwert i.S. eines Vorschlags

Lerngebiet 7: Aufbau und Strukturen von Netzwerken und Netzwerkbetriebssystemen

Zeitrichtwert*: 80 Stunden

Lernziele	Lerninhalte	Hinweise zum Unterricht
7.1 Grundstrukturen von Netzwerken beschreiben	Peer-to-Peer-Netze, Client-Server-Netze	
	Topologien geografische Ausdehnung von Netzwerken	Bus, Ring, Stern, Vermaschung LAN, MAN, WAN
7.2 Strukturmodelle erläutern	OSI-Modell: Schichten, Aufgaben, Protokolle, Kapselung TCP/IP-Protokollstapel, Protokollanalyse	Protokollanalysesoftware einsetzen
7.3 Physikalische Basis der Netzwerktechnik erklären und systematisieren	analoge und digitale Signale, Signalverarbeitung, Übertragungsverfahren, Frequenznutzungen	
	Koaxialkabel, Twisted-Pair-Kabel Glasfaser	Aufbau, Anwendung Kabelparameter (a, NEXT, FEXT), Kategorien, Klassen Fasertypen, Moden, Modendispersion
7.4 Merkmale kollisionsbehäfteter Netze beschreiben	Zugriffsverfahren, Kollisionsdomäne, Frame, MAC-Adressierung, Hub, Repeater	CSMA/CD, CSMA/CA IEEE 802.3
7.5 Segmentierung von Netzen erläutern und begründen	Bridge, Switch, Broadcastdomäne, VLAN	Switch-Arbeitsweise IEEE 802.3q
7.6 WLAN als Übertragungstechnologie erläutern	WLAN, Standards, Netzstrukturen Sicherheitsrisiken, Sicherheitsmaßnahmen	Accesspoint, WLAN-Bridge,
7.7 Logische Strukturierung von Netzen mittels des Internet-Protokolls durchführen	Internet-Protokoll, Adressierung Subnetze, Zweck, Methode,	Subnetzbildung in Klasse A, B, C

7.8	Adresszuordnungsprotokolle beschreiben	ARP, RARP, DHCP	
7.9	Routing erläutern, Routing-Protokolle unterscheiden	Routing, Routing-Protokolle, geroutete Protokolle	Distanz-Vektor-, Linkstate-Routingprotokolle
7.10	Transport-Protokolle erklären	Transport Control Protocol, User Datagram Protocol, Ports	
7.11	Anwendung und Bedeutung von Adressumsetzungen erläutern	Adressumsetzungen	NAT, PAT
7.12	Wichtige netzwerkorientierte Dienste beschreiben	Dienste der Anwendungsschicht	DNS, E-mail, WWW, FTP
7.13	Grundstrukturen von ISDN erläutern	ISDN, NTBA, S ₀ -Bus, Anlagenanschluss, Mehrgeräteanschluss, Terminierung	Visuelle Dokumentation: z.B. Grundriss um ISDN-Verkabelung ergänzen
7.14	Gebäudeverkabelung planen	Verkabelungsprojekt: Strukturierte Verkabelung	Visuelle Dokumentation: z.B. Grundriss um Netzwerkverkabelung ergänzen
7.15	Grundlegende Aufgaben von Netzwerkbetriebssystemen darstellen	Netzwerkbetriebssysteme: Aufgaben, Berechtigungen, Strukturierung von Benutzern, Benutzerverwaltung, Benutzerprofile, Anmeldeskripte	

*Zeitrichtwert i.S. eines Vorschlags