

Abstracts zum 8. ILTB

Amann/Brichzin (Workshop):

Fremde Quelltexte analysieren, verstehen und verbessern

Ein typisches Szenario von Dienstleistern in der Softwarebranche ist, dass Kunden mit einer kranken Software kommen: Das System läuft langsam, verhält sich unerwartet, produziert Fehler. Aufgabe eines Softwareentwicklers ist es dann, das System dynamisch (Verhalten zur Laufzeit) und statisch (Quelltext) zu analysieren. In diesem Workshop wird ein Modul vorgestellt, das Schüler auf die Analyse von (fehlerhafter) Software vorbereitet. Es wird gezeigt, wie Fehler mit Hilfe des Debuggers eingekreist werden können und wie der Quelltext anschließend analysiert werden kann, um die Ursache des fehlerhaften Verhaltens zu finden. Das Modul (2 Unterrichtsstunden) beinhaltet eine kurze Einführung zum Thema Erkennen und Beheben von Softwarefehlern. Daran schließt sich ein größerer Teil mit Übungen an, in denen die Schüler mit Hilfe des BlueJ-Debuggers verschiedenen schwere Fehler in drei Beispielprogrammen finden und korrigieren. Bitte Laptop mit installiertem BlueJ mitbringen.

Bergmann (Workshop):

Die Simulationssoftware Filius im Informatikunterricht der Jahrgangsstufe 12

Mit Filius kann der Punkt 12.2 des Lehrplans (Protokolle, Schichtenmodell, Internet) anschaulich demonstriert werden, ohne dass dazu in Rechner Hardware eingebaut und Leitungen verlegt werden müssen.

Gezeigt wird zunächst die grundsätzliche Funktionsweise des Programms am Beispiel der Verbindung zweier Rechner. Schon hier wird die Rolle der IP-Nummern deutlich und die Aufgaben der Protokollschichten sind dabei gut erkennbar. Danach erfolgt der Aufbau eines Rechnernetzes mit einem Switch, schließlich werden mehrere solcher Netze mit einem Vermittlungsrechner gekoppelt. Um die Funktionsweise des Internets zu demonstrieren kann auch ein Webserver eingerichtet und ein DNS-Server eingebunden werden.

Im Gegensatz zur Realität können hier die übertragenden Daten und die Arbeitsweise der einzelnen Schichten gut verfolgt werden, da die Ablaufgeschwindigkeit variabel ist. Da Vieles ohne Angst vor Beschädigungen einfach und schnell ausprobiert werden kann, unterstützt die Software das entdeckende Lernen.

Die Teilnehmer sollen möglichst einen eigenen Laptop mit installiertem Filius mitbringen.

Ehmann (Vortrag):

Einsatz graphischer Programmierumgebungen bei der Einführung in die Algorithmik im Informatik-Anfangsunterricht

Syntaxprobleme und kryptische Fehlermeldungen, die textbasierte Programmiersprachen bei der Umsetzung von Algorithmen mit sich bringen, sind im Anfangsunterricht oft Hürden für Schüler, die den Blick von den eigentlichen Konzepten der Algorithmik ablenken und demotivierend wirken. Ausgehend von syntaktischen Schülerfehlern zeigt der Workshop Unterschiede bei der Verwendung graphischer Programmierumgebungen im Unterricht auf. Betrachtet werden auch die veränderten Rollen von Modellierung und Programmierung, die sich in der Informatik auch in der modellgetriebenen Softwareentwicklung zeigen. Konkrete Unterrichtsbeispiele mit der graphischen Programmierumgebung Scratch verdeutlichen, wie die algorithmischen Grundstrukturen ausgehend von motivierenden Problemstellungen eingeführt und umgesetzt werden können. Abschließend wird

ein Ausblick auf den MIT App Inventor gegeben: eine graphische Programmierumgebung für Apps. Damit ist es möglich, Apps für Android-Geräte (emuliert und real) zu entwickeln.

Freitag (Vortrag):

Private Daten gegen Kekse – Was ist uns unsere Privatsphäre wert?

Bereitwillig werden persönliche Daten in sozialen Netzwerken, aber auch bei vielen anderen Gelegenheiten preisgegeben. Immer verbunden mit dem Argument „Ich habe nichts zu verbergen!“. Wo aber verläuft die Grenze zwischen einer unproblematischen Teilnahme am modernen Kommunikationsverhalten und an welchen Stellen findet ein massiver Eingriff in die eigene Privatsphäre oder die anderer statt?

Sowohl von offizieller Seite als auch von Eltern und Schülern wird ein verstärkter Einsatz von modernen Kommunikationsmitteln wie Tabletcomputern oder Smartphones und den damit verbundenen Softwarelösungen im Unterricht gewünscht. Damit ist aber auch immer eine Weitergabe von persönlichen Daten verbunden. Wie gehen wir als Lehrer damit um?

Gräßl (Workshop):

Modellierungswerkzeuge für Datenflussdiagramme und Datenbanken

Modellierung spielt im Informatikunterricht eine zentrale Rolle. Dabei ist es für die Schüler besonders gewinnbringend, wenn sie nicht nur mit Papier und Bleistift arbeiten, sondern am Computer eine geeignete Anwendung einsetzen können. In diesem Workshop werden zwei Werkzeuge (für Windows-Betriebssysteme) vorgestellt, die gezielt für die 9. Jahrgangsstufe entwickelt wurden und frei verfügbar sind.

Orinoco: Ein Programm zum Erstellen von Datenflussdiagrammen. Diese können nicht nur gezeichnet, sondern auch anschaulich getestet werden.

YoungDB: Ein Modellierungstool für Klassendiagramme von Datenbanksystemen. Die Entwürfe können ebenfalls sehr einfach getestet werden, da fertige Datenbanken basierend auf Modellierungen exportiert werden können. Zusätzlich beinhaltet YoungDB ein einfaches, aber für den schulischen Bedarf ausreichendes, Datenbanksystem.

Den Teilnehmern wird im Workshop zunächst vermittelt, wie die beiden Werkzeuge arbeiten und eingesetzt werden können. Im zweiten Teil können die Programme, beispielsweise durch das Lösen von typischen Aufgaben aus dem Unterricht, getestet werden. Daraus können sich dann Diskussionen über Möglichkeiten, Grenzen und Weiterentwicklungen basierend auf dem Erfahrungsschatz der Teilnehmer ergeben.

Orinoco: http://klassenkarte.de/?page_id=68

YoungDB: http://klassenkarte.de/?page_id=273

Grillenberger (Vortrag):

Datenmanagement

Datenbanken stellen ein grundlegendes Thema der informatischen Bildung dar. In der Fachwissenschaft Informatik haben in den letzten Jahren verschiedene Entwicklungen auf dem Gebiet "Datenmanagement" stattgefunden, insbesondere die Speicherung und Analyse immer größerer und strukturell vielfältiger Datenmengen unter Zuhilfenahme verteilter Datenbanken („Big Data“). Hingegen konzentriert sich der Informatikunterricht derzeit auf das Thema „Datenbanken und SQL“. Zum Verständnis der Grundlagen moderner Datenspeicherung, wie sie beispielsweise bei Google, Facebook, Twitter oder Amazon stattfindet, reicht dieses Wissen jedoch nicht mehr aus: um die riesigen Datenmengen zu bewältigen müssen diese auf verschiedene Standorte verteilt und

möglichst in Echtzeit verarbeitet werden – während relationale Datenbanken insbesondere auf hohe Datenqualität optimiert sind. Gleichzeitig stellt sich das Problem, dass Daten immer stärker variieren und so kaum ein relationales Schema erstellt werden kann.

Auch Schülerinnen und Schüler haben bei der Verwaltung ihrer privaten Daten mit vielfältigen Aspekten von „Datenmanagement“ zu tun: seien es Redundanz und Konsistenz, die beispielsweise im Kontext von Backup und Synchronisation eigener Daten relevant werden, oder beispielsweise Metadaten, die einerseits Gefahren für die Privatsphäre bergen, andererseits aber in anderen Kontexten (beispielsweise bei der Suche nach Daten) einen großen Mehrwert mitbringen.

In diesem Vortrag werden die aktuellen Entwicklungen im Datenmanagement mit einem Fokus auf möglicherweise für die Schule relevanten Aspekte betrachtet und gemeinsam mit verschiedenen Beispielen vorgestellt sowie die Auswirkungen auf den Unterricht verdeutlicht. Auch Zusammenhänge mit aktuellen gesellschaftlichen Entwicklungen werden einbezogen.

Hennecke (Vortrag):

Modellvorstellungen für das Internet

Immer wieder fallen bei Schülern – aber auch bei Erwachsenen – gravierende Fehlvorstellungen über den Aufbau des Internets auf. Beispielsweise glauben viele Schüler an eine sternförmige Topologie, an dessen Spitze „der“ Server zu finden oder selbst bekannte Politiker daran, dass eine innerhalb Deutschland versendete E-Mail das Land während ihres Transports niemals verlässt. Das zweite Beispiel zeigt, wie schnell in Zeiten zunehmender Überwachung des Netzes derartige Fehlvorstellungen zum Verlust der informativen Selbstbestimmung beitragen. Da sich tragfähige Vorstellungen offensichtlich nicht durch alltägliche Nutzungserfahrungen bilden, sind verständliche Modellvorstellungen früher oder später von unterrichtlicher Relevanz. Der Vortrag stellt dazu passende und gut vermittelbare Modellvorstellungen für unterschiedliche didaktische Reduktionsstufen vor und zeigt ihre Möglichkeiten aber auch ihre Grenzen auf – und wer weiß, vielleicht haben auch Sie danach ein erweitertes Modell vom Aufbau des Netzes der Netze.

Heuer/Pfeffer (Vortrag):

Mobile Computing / Physical Computing - was bringt's? (35min, Heuer)

Der Vortrag stellt einige Überlegungen -zum ereignisorientierten Paradigma -zur Eingabe via Sensoren -zu didaktischen grafischen Programmiersprachen und zu textbasierten Sprachen vor und reflektiert diese an Beispielen.

Interessant sind diese Gedanken für Informatiklehrkräfte, die ab der 8.Jahrgangsstufe (z.B. Wahlkurse) bzw. in der Q-Phase (z.B. Seminare) unterrichten.

Ute Heuer, Universität Passau

Module für den Einsatz in der Q-Phase oder im Wahlunterricht (35min, Pfeffer)

Es werden drei Module vorgestellt, die im Rahmen eines Kooperationsprojekts von "TfK - Technik für Kinder e.V." und Universität Passau entstanden sind:

* Mobile Devices programmieren: - Werkzeug AppInventor - Werkzeug Java-Android

* Physical Computing: - Werkzeug Raspberry Pi

Zu jedem Modul wurde eine umfangreiche Ausarbeitung erstellt, die Lehrkräften das Umsetzen des Moduls im Wahlunterricht oder in Seminaren der Q-Phase erleichtert. Die ersten beiden Module zeigen jeweils exemplarisch, wie eigene Apps programmiert werden können. So können beispielsweise Beschleunigungs- und GPS-Daten ausgelesen werden, es kann persistent gespeichert werden, einmal mit Hilfe einer didaktischen, grafischen Umgebung mit begrenzten Möglichkeiten, einmal mit einer textbasierten professionellen Umgebung. Das dritte Modul "Physical Computing" lotet einfache Umsetzungsmöglichkeiten an der Schnittstelle Informatik / Elektronik aus. Schaltungen können selbst zusammen gebaut und an den Einplatinencomputer Raspberry-Pi

angeschlossen werden. Kleine Python-Skripte, die Spannung an einigen Stellen der Schaltung anlegen und an anderen Stellen der Schaltung messen, können am Einplatinencomputer geschrieben und getestet werden.

Jeanrenaud/Scheibl (Workshop):

Gendersensibilisierung in der Informatik

Geschlechterverhältnisse bleiben im Alltag oft unhinterfragt, weil sie als „natürlich“ angesehen werden. Der Workshop soll die Teilnehmer/innen für Gender- und Diversityaspekte in der Kommunikation sensibilisieren. Die Schwerpunkte liegen dabei auf der gendersensiblen Kommunikationsprozessgestaltung, auf dem gendergerechten Marketing und der gendergerechten Didaktik.

Kahler (Workshop):

3D-Drucker

Der FabLab München e.V. eine offene Hightechwerkstatt stellt die vielfältigen Einsatzmöglichkeiten von computergesteuerten Maschinen wie 3D Drucker, 3D Scanner, Lasercutter, Schneideplotter, CNC Fräse und Elektronik u.a. in der Schule vor. Es wird speziell zu 3D Druck auch einen kleinen Mitmachworkshop geben, in dem verschiedene kostenfreie 3D Modellierungsprogramme und ein 3D Scanner vorgestellt werden und ausprobiert werden können. Ausgewählte Objekte können auch an einem Ultimaker gedruckt werden.

Kastl (Vortrag):

Agile Softwareentwicklungsprojekte in der Schule

Historisch gewachsen orientieren sich Softwareprojekte bislang meist am Wasserfallmodell. Doch daraus erwachsen viele Probleme - für Lernende ebenso wie für Lehrende. Eine gute Projektplanung ist fast unmöglich, weil am Jahresende viele Stunden ausfallen. Es gilt eine lange Modellierungsphase zu motivieren, danach betreibt man "Turnschuhdidaktik" um alle Best möglich zu unterstützen und am Ende ist zwar was codiert, aber nutzbar ist die Software nicht, weil die Zeit zum Testen und Fehler beheben fehlt. Auch Softwareentwickler berichten von Termin- Qualitäts- und Motivationsproblemen. Doch in der IT-Brache hat sich im letzten Jahrzehnt einiges getan - agile Softwareentwicklung ist heute im professionellen Bereich nicht mehr weg zu denken. Was macht sie anders und was bringt sie uns in der Schule? Wir stellen euch ein Set an agilen Praktiken und Techniken vor, das wir zusammen mit interessierten Lehrkräften entwickeln. An einem konkreten Projekt skizzieren wir, wie diese Praktiken beispielsweise zusammenspielen können und wie agile Softwareentwicklung in der Schule ablaufen kann, welche Überlegungen dahinter stecken und welche positiven und hochmotivierenden Erfahrungen die beteiligten Lehrkräfte bislang gemacht haben.

Romeike (Workshop):

Smart und reich durch App-Entwicklung - Programmieren in der Sekundarstufe mit App Inventor

Handys und Smartphones stellen für viele Jugendliche einen wichtigen Gegenstand im Alltag dar. Für viele Softwarefirmen besteht ein Großteil des Geschäfts inzwischen aus der Entwicklung mobiler Applikationen für Smartphones. Während das Entwickeln solcher Apps für Laien bisher

noch zu komplex war, bietet App Inventor eine in ihrer Komplexität reduzierte Möglichkeit an, per Baustein-Programmierung Applikationen für Android-Smartphones zu entwickeln. Die Programmierumgebung orientiert sich dabei an Scratch. Applikationen können in Echtzeit auf einem angeschlossenen Smartphone oder dem mitgelieferten Emulator ausprobiert werden. Schnittstellen wie GPS, Beschleunigungssensor oder Webzugriff können einfach eingebunden werden. Selbst die Veröffentlichung der Applikationen im Android-App-Market ist möglich. Im Workshop werden Grundlagen, Möglichkeiten und Grenzen dieser Android-App-Entwicklung in der Schule besprochen sowie einfache Applikationen programmiert. Teilnehmer sind nach dem Workshop in der Lage, selbst Applikationen zu entwerfen und Unterrichtsideen mit App Inventor zu entwickeln. Der Einsatz von App Inventor in der Schule als Werkzeug verspricht smarte Schüler, die mit der richtigen Idee durchaus reich werden können - vielleicht nicht nur reich an Erfahrung...
Informationen zu App-Inventor http://www.informatiktools.de/wiki/index.php5?title=App_Inventor

Ruf (Workshop):

Aufgaben unter die Lupe genommen

Aufgaben spielen in Informatik eine zentrale Rolle, sei es als Beispiel- oder Übungsaufgaben im Unterricht, als Hausaufgaben zu Hause oder als Testaufgaben in Exen oder Schulaufgaben. Wir Lehrkräfte erstellen diese Aufgaben oder wählen sie aus einer Aufgabensammlung aus, korrigieren die Lösungen unserer Schüler und nutzen Aufgaben zur Evaluierung unseres Unterrichts. Worauf sollte man beim Erstellen von Aufgaben achten? Wie lassen sich Aufgaben variieren oder deren Schwierigkeitsgrad einschätzen? Welche Aufgabentypen finden im Informatikunterricht überhaupt Verwendung? Diese und ähnliche Fragen wollen wir in diesem Workshop behandeln und dabei gemeinsam Aufgaben erstellen, analysieren und besprechen. Ziel ist es, einen geschärften Blick für Aufgaben zu entwickeln, der Ihnen den Umgang mit ihnen im Unterrichtsalltag erleichtern soll.

Schwaiger (Vortrag):

Aktuelles aus dem ISB

„LehrplanPLUS steht für ein umfangreiches Lehrplanprojekt, in dem zeitgleich und inhaltlich abgestimmt die Lehrpläne für alle allgemein bildenden Schulen sowie die Wirtschaftsschulen und die beruflichen Oberschulen überarbeitet werden.“

(<https://www.lehrplanplus.bayern.de/seite/lehrplanplus>)

Auch die Lehrpläne des Gymnasiums erfahren im Zusammenhang dieses Projekts eine kompetenzorientierte Ausrichtung. Im Rahmen des Vortrags soll hiervon ein Einblick gegeben und - soweit derzeit möglich - auf fachspezifische Aspekte, insbesondere das Kompetenzstrukturmodell für den Lehrplan Informatik, eingegangen werden.

Shah (Vortrag und Workshop):

Fehlvorstellungen in der objektorientierten Programmierung

Zu dem Thema „Fehlvorstellungen objektorientierter Programmierung in der Schule“ findet man in der aktuellen Forschung nicht viel. Studien liegen entweder Studenten fortgeschrittenen Semesters als Datenlage zugrunde oder die Erhebungszahlen reichen für gesicherte Aussagen nicht aus. Zuerst werden Ergebnisse einzelner wissenschaftlicher Veröffentlichungen zu diesem Thema vorgestellt. Anschließend wird zu diesem Themenpunkt praktisch und auf die Schule bezogen am Computer gearbeitet.

Steinert (Vortrag):

Fragen des Datenschutzes im Informatikunterricht der 6. und 7. Jahrgangsstufe

Der Datenschutz ist im aktuellen Lehrplan zur Informatik innerhalb des Faches Natur und Technik nur in sehr geringem Maße enthalten. In der Jahrgangsstufe 6 sollten z.B. Fragen des Urheberrechts beim Erstellen von Multimediadokumenten berücksichtigt werden und in Jahrgangsstufe 7 müssten beim Erarbeiten vernetzter Dokumente auch rechtliche Aspekte berücksichtigt werden. Die vor einiger Zeit vom ISB durchgeführte Umfrage zum derzeitigen Lehrplan zeigt, dass bei den Kolleginnen und Kollegen diesbezüglich fachliche Unsicherheiten und didaktischer Nachholbedarf bestehen. Darüber hinaus ist das Thema Datenschutz durch das Verhalten von Firmen wie Facebook oder Google, oder der NSA mit Vehemenz in den Blick der Öffentlichkeit gerückt. Die Schule und insbesondere der Informatikunterricht wird sich diesen Fragen auf die Dauer nicht entziehen können.

Dieser Vortrag konzentriert sich auf die Grundlagen urheberrechtlicher Fragen und der wesentlichen Persönlichkeitsrechte im Hinblick auf den Unterrichtseinsatz für die Jahrgangsstufen 6 und 7. Dabei werden zunächst die rechtlichen Grundlagen vorgestellt. Anschließend erfolgt eine Elementarisierung für den Einsatz im Informatikunterricht und Skizzen möglicher Unterrichtsabläufe werden präsentiert. Aufgaben- und Lösungsvorschläge zu den Themenbereichen Urheberrecht und Persönlichkeitsrechte runden den Vortrag ab.

Voß (Workshop):

Einsatz visueller Programmierumgebungen im Informatikunterricht am Beispiel von Scratch/Snap!

Scratch bzw. *Snap!* sind visuelle Programmiersprachen, mit denen man mit relativ wenig Aufwand ansprechende Programme gestalten kann. Beide Programme sind direkt im Browser mit Internetzugang lauffähig. Im Informatikunterricht der Unterstufe des Gymnasiums bietet sich der Einsatz von *Scratch* als einfaches Programmierwerkzeug zur Simulation und Umsetzung von Abläufen mit grundlegenden Algorithmen an. *Snap!* erweitert *Scratch* um abstraktere Konzepte, wie Rekursion oder Listen. Bei der Behandlung informatischer Themen in höheren Klassen, bei der auf eine textuelle Implementierung verzichtet wird, kann *Snap!* als Erweiterung zur Umsetzung von reinem Pseudocode eingesetzt werden. Im Workshop werden einige Beispiele für den Einsatz von *Scratch* bzw. *Snap!* vorgestellt. Anhand gestellter Aufgaben erhalten die Teilnehmer die Möglichkeit, selber praktisch zu arbeiten.

Walter (Workshop):

Animation mit GIMP

Mit der beliebten Open Source Software GIMP kann man, neben den zahlreichen Möglichkeiten zur Bildbearbeitung, auch Animationen erstellen. Die Möglichkeiten reichen hier von der Gif-Animation bis zum "Morphen". Der Workshop beschäftigt sich mit eben diesen Möglichkeiten, es werden selbst verschiedene Animationen erstellt.

Teilnahmevoraussetzungen: Die Workshopteilnehmer bringen ihr eigenes Notebook mit installiertem GIMP (bestenfalls in der Version 2.8) mit.

Weishaupt (Workshop):

Android Programmierung mit Java – Ein schneller Einstieg

Applikationen für Smartphones und Tablets werden heute von vielen Jugendlichen eingesetzt. Die große Motivation seitens der Schüler kann z.B. im Rahmen eines P-Seminars oder im Informatikunterricht genutzt werden, um neue Möglichkeiten im Unterricht zu eröffnen.

Modernes Design und innovative Ideen zur Bedienoberfläche bilden zusammen mit der Programmiersprache Java die Grundlage für die Entwicklung von eigenen Android-Apps.

Der Workshop wird das grundlegende Konzept der Programmierung von Android Apps an einem Beispiel vorstellen. Um die notwendigen Schritte selbst nachvollziehen zu können, sind gute Kenntnisse der Programmiersprache Java notwendig.

Wenn Sie selbst die Programmierung im Workshop nachvollziehen möchten, benötigen Sie außerdem einen eigenen Rechner mit installiertem Android Studio und zugehörigem Android SDK. Eine Anleitung zur Installation finden Sie unter <https://developer.android.com/sdk/index.html?hl=i>.

Wiedemann (Workshop):

iOS-Programmierung mit Swift und Xcode-Playground: Eine Alternative zu Java/BlueJ

Der Workshop zeigt neue Möglichkeiten auf, den Informatikunterricht ab der 10. Jahrgangsstufe auf Apple-Computern umzusetzen. Insbesondere werden angesprochen: Die neue Programmiersprache Swift (im Vergleich mit Java), die interaktiven Möglichkeiten von Xcode-Playground als Entwicklungsumgebung sowie die sehr einfache Möglichkeit, die entwickelten Produkte auf IOS-Geräten (iPad, iPhone) auszuführen. Als Beispiele werden die Leitbeispiele aus dem Oldenbourg-Buch für die 10. Klasse verwendet.