

Betz, H.

## Berufsbegleitende Lehre im Fach Algorithmen und Programmierung

Das Fach Algorithmen und Programmierung stellt eine essentielle Grundlagenveranstaltung im Bereich der technischen Informatik für die Ingenieurstudiengänge dar. Für das berufsgleitende Bachelorstudium wurden Inhalt und Struktur der Präsenzveranstaltung beibehalten und intensiv auf Online-Übungsaufgabenunter anderem mithilfe eines Programmierertools zu automatischen Auswertung von Programmcode gesetzt. Das positive Feedback zu angepassten Lehrmethoden bestätigt den eingeschlagenen Weg. Viele der für den berufsbegleitenden Modus entwickelten Verfahren können auch in den Präsenzmodus übertragen werden, so dass die Lehre im Fach Algorithmen und Programmierung zukünftig flexibler gestaltet werden kann.

### Inhalt

1. Rahmenbedingungen .....	1
2. Berufsbegleitendes Studium: Anforderungen .....	1
3. Veranstaltungen: Ziele, Inhalte, Methoden .....	1
4. Präsenz- und berufsbegleitendes Studium im Vergleich .....	2
4.1. Inhaltliche Unterschiede zwischen Präsenz- und Berufsbegleitendem Studium .....	2
4.2. Zeitliche Betrachtung von Präsenz- und Berufsbegleitendem Studium .....	3
5. Ergebnisse .....	4
5.1. Qualitative Ergebnisse .....	4
5.2. Quantitative Ergebnisse .....	4
6. Fazit .....	4
7. Ergänzungen zu VPL .....	5
7.1. VPL .....	5
7.2. Entstehung und Probleme des VPL-Systems .....	5
Autor .....	6
Hinweise .....	6

### 1. Rahmenbedingungen

Die Lehrveranstaltung „Algorithmen und Programmierung“ (AuP) sowie das hiermit zusammenhängende „Praktikum Praktische Informatik“ (Praktikum PI) wurde gemäß dem BASICplus-Lehrmodell mit zwei Studierenden erprobt. Neben den beiden Pflichtfächern wurde ein, von den Studierenden gewünschtes Lehrangebot „Praktikum Arduino“ ergänzend durchgeführt.

Beide Teilnehmer studieren im berufsbegleitenden Modus und werden von ihren Arbeitgebern für die Präsenzveranstaltungen am Freitag freigestellt. Neben den Freitagsterminen werden alle vier Wochen die Samstage für weitere Lehrveranstaltungen hinzugenommen. Eine Lehrinheit entspricht 90 Minuten. Insgesamt sind für alle Lehrveranstaltungen des Faches AuP 90 Stunden Bearbeitungszeit der Studierenden vorgesehen (34 Stunden Präsenz, 56 Std. Selbststudium). Einerseits wurden in großem Umfang vorhandene Lehrmaterialien und Tools genutzt und didaktisch an die spezifischen Erfordernisse der BASICplus-Studierenden angepasst. Außerdem wurden neu entwickelte Lehrinheiten mit diesen Studierenden bearbeitet.

### 2. Berufsbegleitendes Studium: Anforderungen

Im Vergleich zum Präsenzstudium ist die Erfüllung folgender Anforderungen bei der berufsbegleitenden Lehre zwingend notwendig:

- Die verwendeten Vorlesungsmaterialien müssen vor der Veranstaltung bereits online gestellt und jederzeit abrufbar sein. Hierunter zählen die Vorlesungsskripte und Übungsaufgaben.
- Eine interaktive Unterstützung zur Lehre wird gefordert, wie z.B. eine direkte Rückkopplung zu den Studierenden über die Präsenzveranstaltung hinaus.

### 3. Veranstaltungen: Ziele, Inhalte, Methoden

Die Inhalte und Ziele der Vorlesung und Übung sind im Fachleitfaden, der Modulbeschreibung und Fachbeschreibungen der Studiengänge (siehe [http://www.tu-ilmenau.de/modultafeln/Maschinenbau/Bachelor/2013/modul/11155/?no\\_cache=1](http://www.tu-ilmenau.de/modultafeln/Maschinenbau/Bachelor/2013/modul/11155/?no_cache=1)) beschrieben.

Beide Veranstaltungen werden regulär im 1. Fachsemester der eingebundenen Ingenieurstudiengänge angeboten. Die wichtigsten Inhalte der Veranstaltung sind wie folgt:

- Einführung in Algorithmen
- Grundkonzepte der Programmierung
- Entwurf von Algorithmen
- Algorithmenparadigmen
- Eigenschaften von Algorithmen
- Suchen und Sortieren
- Algorithmenmuster
- Abstrakte Datentypen und Objekt-orientierte Programmierung
- Parametrisierte Datentypen
- Hashverfahren

Ein spezielles Bonussystem wurde für das Fach AuP eingesetzt um

- die Motivation der Studierenden zum kontinuierlichen und tiefgründigen Bearbeiten des Lehrstoffes zu fördern,
- die Präsenzzeit durch gute Vorbereitung der Studierenden effektiv nutzen zu können sowie
- durch Auswertung der eingereichten Lösungen, Feedback zum Leistungsstand der Studierenden zu erhalten.

Während des Semesters kann jeder Studierende durch das Lösen von Bonusaufgaben bis zu 12 Prozentpunkte zusätzliche Klausurpunkte als Bonus erreichen. Die Bonuspunkte werden jedoch nur hinzugerechnet, sofern die Klausur mindestens mit einer vier bestanden wurde. Die Bonusaufgaben sind über drei verschiedene Übungsblätter verteilt. Jedes Übungsblatt behandelt einen anderen Themenschwerpunkt, jede Übungsaufgabe eine andere Nuance desselben Themas. Der Prozess zum Erreichen der Bonuspunkte ist dreigeteilt. Nach der Ausgabe des Übungsblattes (Teil 1) müssen die Studierenden innerhalb von zwei Wochen einen Lösungsvorschlag via Moodle bereitstellen. In der korrespondierenden Übung (Teil 2) werden Probleme angesprochen, Fragen geklärt sowie inhaltliche ähnliche Aufgaben behandelt (Transferaufgaben). Nach einer weiteren Woche müssen die in Teil 1 bereitgestellten Lösungen verbessert und nochmals via Moodle bereitgestellt werden. Die Bewertung und Verteilung der Bonuspunkte wird unterstützt durch Nutzung des Virtuellen Programmierlabors (VPL, <https://pi.tu-ilmeneau.de>) des Fachgebietes Datenbanken und Informationssysteme. In Teil 1 und 3 kann der Studierende jederzeit auf das vollautomatische Testsystem zugreifen und seine Lösung entsprechend anpassen. Zur Unterstützung der Einweisung in die Handhabung des VPL entstanden Videos und Beispiele, die sich mittlerweile auch bei der Nutzung durch die Präsenzstudierenden bewährt haben.

Das Praktikum „Praktische Informatik“ wird zum Bestehen des Moduls gefordert und ist regulär im 2. Semester angesiedelt. In diesem wird das theoretisch erworbene Wissen in einer praktischen Arbeit überprüft.

Das Praktikum „Arduino“ ist noch nicht für alle Studierenden im regulären Lehrplan eingebunden. Vergleichbare Lehrangebote wurden in den letzten Jahren im Rahmen der Erprobung des BASIC-Lehrmodells mit mehreren Studiengruppen entwickelt und erprobt (siehe weiter unten).

Auf Wunsch der Studierenden wurde ein inhaltlich ähnliches Praktikum vorbereitet und als Ergänzungslehrrangebot durchgeführt. In diesem wird das theoretisch und praktisch erworbene Wissen aus den vorherigen Veranstaltungen nochmals anschaulich wiederholt und auf ein anderes Themengebiet übertragen.

#### 4. Präsenz- und berufsbegleitendes Studium im Vergleich

##### 4.1. Inhaltliche Unterschiede zwischen Präsenz- und Berufsbegleitendem Studium

Die Vorlesung und Übung sind vollständig äquivalent zum Angebot im Präsenzstudium. In der ersten zwanzigminütigen Veranstaltung müssen die Studierenden einen Fragentest bestehen. In diesem werden theoretische und programmiertechnische Fragen zu Java gestellt. Zu jeder Frage sind drei mögliche Antworten gegeben, aus denen der Studierende die jeweils Richtige(n) auswählen muss. Die Bewertung wird automatisch vom PC (Nutzung des integrierten Testsystems im VPL) vorgenommen. Im zweiten sechzigminütigen Teil wird eine den Studierenden vorher nicht bekannte Programmieraufgabe gestellt, die mit den in der Vorlesung vermittelten Kenntnissen gelöst werden soll. Diese muss in Java umgesetzt und mit einem automatisierten Testsystem auf korrekte Ergebnisse überprüft werden.

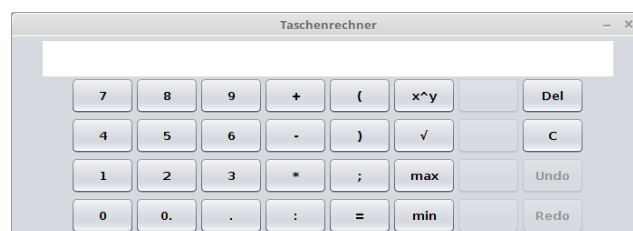


Abb. 1: Oberfläche des für die Programmieraufgabe herangezogenen Taschenrechners

In der Umsetzung mit den BASICplus-Studierenden wurde das AuP-Praktikum auf Anregung der Studierenden abgeändert. Seitens der Studierenden bestand Interesse an einer praxisrelevanten Anwendungsaufgabe, die in ein größeres Java-Projekt eingebunden ist. Ziel ist es, die dabei benötigte Strukturierung in Packages, Klassen und Methoden besser zu verstehen und die internen Abläufe nachvollziehen zu können. Desweiteren sollte die Objektorientierung

an einem Beispiel veranschaulicht werden. Hierfür wurde vom Lehrenden eine Beispielanwendung „Taschenrechner“ implementiert. Dieser unterstützt alle Grundrechenarten sowie weitere Operationen. Weiterhin werden beliebige und ineinander geschachtelte Klammerpaare unterstützt. Intern ist eine Grammatik definiert, die einem Parser für die Auswertung der Benutzereingaben nutzt und daraus einen Baum aufbaut. Das Frontend, mit allen Buttons, wurde von den Benutzeraktionen und dem Backend streng getrennt. Die Interaktion zwischen den Schichten wird über das Observer-Pattern und Abstraktionen gelöst.

Der genaue Aufbau sowie die internen Abläufe wurden in den ersten beiden Lehreinheiten anhand einer Präsentation genauer erklärt und mit Beispielen illustriert. Hierfür wurde die Modellierungssprache UML neu eingeführt. Der Quellcode sowie die Präsentation wurden nach der Lehrveranstaltung sofort via Moodle bereitgestellt.

Vier Wochen später fand das eigentliche Praktikum in zwei weiteren Lehreinheiten statt. Hierfür wurden mehrere, den Studierenden vorher nicht bekannte und aufeinander aufbauenden sowie im Schwierigkeitsgrad steigende, Aufgaben ausgegeben, die die Studierenden einzeln lösen und in der daran folgenden Abnahme erläutern mussten.

**Aufgaben:**

1. Im Frontend sollten die leere Beschriftung eines Buttons mit einer Beschriftung versehen werden.
2. Im Frontend sollten zwei Buttons getauscht werden.
3. Im Backend sollte mindestens ein neuer Operator (sin oder cos) hinzugefügt werden.
4. Im Frontend sollten ein leerer Button beschriftet und im Backend ein neuer Operator hinzugefügt werden.
5. Es sollte eine Historie aller bereits getätigten Aktionen angelegt und ein Undo sowie Redo ermöglicht werden.

Jede Aufgabe wurde einzeln abgenommen und war mit Fragen zur jeweiligen Implementierung verbunden.

Das Praktikum „Arduino“ basiert auf dem von der „Basic Engineering School“ entwickelten Autonomen Miniaturtransporter (AMT). Dies ist ein kleiner real existierender Roboter, der mittels Helligkeitssensoren, die zwischen den Rädern angebracht sind, eine schwarze Linie auf einem Wegeplan folgen kann. Die Sensorwerte werden von einem Arduino entgegengenommen, ausgewertet und in Aktorenbefehle, wie der Motorsteuerung, übersetzt. Die Programmierung des Arduinos geschieht in C bzw. C++ und ist sehr ähnlich zu der in der Vorlesung vorgestellten Sprache Java. Das Praktikum begann mit einer theoretischen Einführung zum Arduino und dessen Möglichkeiten sowie Beschränkungen. Daran anschließend sollte eine LED blinken, was zu einem Lauflicht weiter ausgebaut wurde. Im nächsten Schritt soll der AMT vorwärts bzw. rückwärts fahren. Nun wurde zu einem Simulator gewechselt, der vergleichbare physikalischen Eigenschaften aufweist wie ein real existierendes AMT. In diesem wurde die Linienverfolgung

implementiert und mit einer Kreuzungserkennung mit Abbiegevorrichtung versehen. Schließlich wurde eine einfache Wegeplanung umgesetzt und in den bestehenden Code eingebunden.

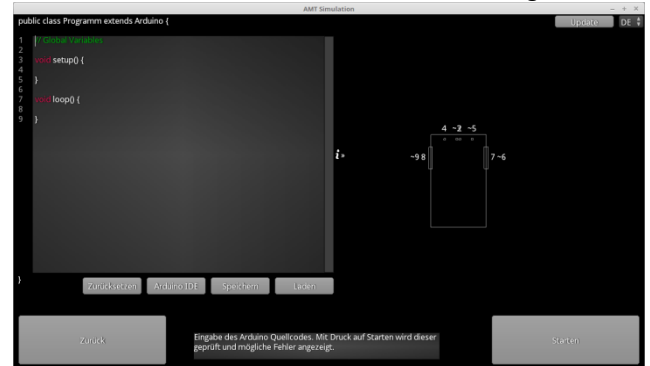


Abb. 2: Programmierumgebung des Arduino



Abb. 3: Wegeplan und Simulationsumgebung

#### 4.2. Zeitliche Betrachtung von Präsenz- und Berufsbegeleitendem Studium

Durch die strenge Planung in Semestern, die klar definierten Lernziele, die mit dem Präsenzstudium jederzeit vergleichbar sein müssen und die Beschränkung auf durchschnittlich 1,25 Präsenz-Vorlesungstage pro Woche, können nur wenige Fächer pro Semester vollständig abgehandelt werden. Insbesondere ist es für die Studierenden eine zusätzliche Belastung, in nur wenigen Wochen den gesamten Stoff einer 15-wöchigen Vorlesung vollständig zu durchdringen, was zu einem erhöhten Anteil an Selbststudium und zusätzlicher Arbeitszeit führt. Da beide Teilnehmer voll berufstätig sind (mit teilweise 40 Stunden Arbeitszeit pro Woche), was zu Konflikten führt. Auch erfordert es von den Lehrenden eine intensivere Vorbereitung, sich auf die wichtigsten Teile zu konzentrieren, vorhandenes (Vor)Wissen gut einzuschätzen darauf aufbauendes Wissen entsprechend zu vermitteln. Gleichzeitig muss das vermittelte Wissen jederzeit mit dem Präsenzstudium vergleichbar sein. Aus diesem Grund hat die Veranstaltung „Algorithmen und Programmierung“ fast denselben Umfang wie im Präsenzstudium.

Die Präsenzveranstaltung besitzt 15 Vorlesungen und 7 Übungen (22 Lehreinheiten plus 1 Lehreinheit für die schriftliche Prüfung).

Insgesamt wurden in diesem Projekt 22 Lehreinheiten für das Fach aufgewendet, wobei die letzte Einheit mit einer schriftlichen Prüfung belegt wurde. Eine klare Trennung in Vorlesung und Übung hat sich bei der sehr kleinen Teilnehmerzahl als schwierig erwiesen, so dass beide Veranstaltungen eng miteinander verzahnt wurden.

Das Praktikum „Praktische Informatik“ ist in der Präsenzveranstaltung mit einer Lehreinheit Präsenzzeit für die Studierenden eingeplant. Im Projekt wurden vier Lehreinheiten Präsenzzeit aufgewendet. Das zusätzlich angebotene Praktikum „Arduino“ wurde mit weiteren vier Lehreinheiten betreut.

## 5. Ergebnisse

### 5.1. Qualitative Ergebnisse

Beide Studierende haben die Prüfung erfolgreich absolviert. Die sehr enge Verzahnung von Vorlesung und Übung hat sich als sehr positiv für die Vermittlung von Lehrinhalten herausgestellt. Die Umsetzung der Verzahnung von Vorlesung und Übung wurde durch die überschaubare Gruppengröße und die ständige Verfügbarkeit des Computelabors zu den Lehrveranstaltungszeiten begünstigt.

Das AuP-Praktikum wurde von beiden erfolgreich absolviert. Positiv anzumerken ist die intensive Vorbereitung. Beide haben den ausgegebenen Programmcode mit der dazugehörigen Präsentation intensiv studiert und teilweise auch modifiziert. Die Aufgaben wurden von beiden schnell, zielstrebig und ohne große Probleme gelöst. Auch wurde die Motivation durch die abgeänderte Form des Praktikums stets aufrechterhalten.

Die Rückmeldung der Studierenden war größtenteils positiv. Insbesondere die Berücksichtigung des individuellen Wunsches eines größeren Projektes wurde sehr positiv aufgenommen. Die Herangehensweise in Präsentationstermin mit Demonstration, vier Wochen Zeit zur Einarbeitung und anschließenden Praktikum wurde sehr gelobt. Weniger positiv wurde von den Studierenden die Menge an Quellcode zur Lösung des Problems eingeschätzt. Jedoch bedingt ein größeres Projekt in der Praxis mehrere tausend Zeilen Quellcode. Der Taschenrechner umfasst mit allen Kommentaren 2450 Zeilen.

Das unbenotete „Praktikum Arduino“ startete mit dem Ziel, eine einfache Linienverfolgung zu implementieren. Dieses Ziel wurde bereits nach der Hälfte der offiziellen Praktikumszeit erreicht, wodurch weitere Aufgaben bearbeitet werden konnten.

Sehr positiv ist die sehr schnelle und größtenteils fehlerfreie Überführung eines Problems in ein fertiges Programm. Beide Studierenden konnten ihr zuvor erworbenes

Wissen auf ein neues Problemfeld übertragen und festigen. Die Rückmeldung der Studierenden war sehr positiv und von eigener Initiative geprägt. Beide möchten sich einen Arduino anschaffen und diesen für private Projekte nutzen.

### 5.2. Quantitative Ergebnisse

Die Noten der schriftlichen Prüfung liegen im Durchschnitt im Vergleich zu den Noten der Präsenzveranstaltung. Bei näherer Betrachtung der Klausur ist weiterhin kein Unterschied zu der Präsenzveranstaltung der regulär Studierenden festzustellen. Beide Teilnehmer haben im Theorieteil der Klausur wesentlich besser abgeschnitten als im Programmiereteil. Dieses Ergebnis ist mit der Klausur der regulär Studierenden vergleichbar.

Die soeben gemachten Aussagen sind wegen der geringen Teilnehmerzahl keine statistisch validen Ergebnisse, eher eine Tendenz.

Durch die abgeänderte Form und der sehr geringen Teilnehmerzahl des AuP-Praktikums ist keine direkte quantitative Aussage möglich.

## 6. Fazit

Eine reine zeitliche und quantitative Betrachtung zwischen Präsenz- und berufsbegleitendem Studium ist nicht zielführend (mehr Lehrleistung bei gleichbleibendem Ergebnis). Hier ist insbesondere die qualitative Betrachtung wichtig. Es ist erkennbar aufgefallen, dass beide Studierende sich intensiv in die Praktika eingebracht haben. Zudem sind die Programmierkenntnisse beider Teilnehmer besser als in der Vergleichsgruppe (Präsenzstudium). Dies war insbesondere beim Praktikum „Arduino“ erkennbar. Beide Studierende hatten in kurzer Zeit das gesetzte Ziel erreicht.

Ein Grund kann die geringe Gruppengröße sowie die damit verbundene bessere und zielgerichtete Unterstützung bei Problemen und Fragen sein. Auch spielt die gesteigerte Motivation eine zentrale Rolle. Beide Teilnehmer werden explizit von ihren Arbeitgebern für einen Tag pro Woche freigestellt.

Neben der Vergleichsgruppe der regulär Studierenden kann eine weitere Modellgruppe der „Basic Engineering School“ mit einbezogen werden. Ein Ziel des BASIC-Lehrmodells ist es, die Vermittlung von interdisziplinärem Wissen zu unterstützen und für die Studierenden eine bessere und praxisorientiertere Ausbildung zu ermöglichen, was sich nachweislich in einem besseren Notendurchschnitt von 0,5 bis zu einer Gesamtnote und einer geringeren Abbrecherquote von bis zu 25 bis 30 % widerspiegelt. Anhand der bisherigen Erfahrungen im BASICplus als auch im BASIC-Lehrmodell sind die qualitativen Erfolge beider Ansätze sehr ähnlich und übertragbar. Deshalb arbeitet die TU Ilmenau daran, mit den Erfahrungen beider Modelle in das allgemeine Lehrangebot weiter zu entwickeln.

## 7. Ergänzungen zu VPL

### 7.1. VPL

Bereits zu Beginn des Artikels wurde das VPL-System erwähnt. Dies ist ein selbst entwickeltes System zur automatischen Überprüfung von Java-Programmcode. Hierfür muss der von den Studierenden entwickelte Programmcode über ein Web-basiertes Frontend hochgeladen werden. Das System nimmt die Anfrage entgegen und überprüft den Quellcode nach verbotenen Konstrukten. Dies sind Sprachelemente, die explizit für die Problemlösung ausgeschlossen sind, wie z.B. fertige Lösungen/Bibliotheken wie `Math.min()` aus der Java-Bibliothek zur Bestimmung des Minimums. Daraufhin wird der Quellcode kompiliert. Tritt ein Kompilierfehler auf, werden alle nachfolgenden Schritte abgebrochen und eine Fehlermeldung erscheint. Nun wird der Quellcode mit einer zuvor entwickelten und vom Lehrenden bereitgestellten Unit-Test-Klasse getestet. Hierfür werden alle vom Studierenden entwickelten Methoden sequenziell abgearbeitet und mit den erwarteten Ergebnissen verglichen. Tritt eine Abweichung auf, wird eine entsprechende Fehlermeldung auf dem Bildschirm ausgegeben. Anderenfalls wird der Test als bestanden gewertet.

Dieses System wurde während der gesamten AuP-Übung als zusätzliche Lehrform eingesetzt. In der Präsenzlehre wird es im Praktikum „Praktische Informatik/Teil Algorithmen und Programmierung“ seit mehreren Jahren erfolgreich verwendet.

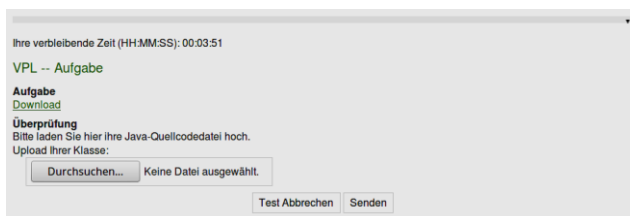


Abb. 4: Screenshot des VPL-Systems

### 7.2. Entstehung und Probleme des VPL-Systems

Das VPL-System ist auf Anregung der Studierenden vor einigen Jahren (2012) entstanden. Es wurde die „subjektive“ Bewertung von Bonuspunkten sowie des Praktikums „Praktische Informatik“ als Anlass genommen. Seitdem wird die Bewertung und Vergabe von Punkten für das Praktikum durch das System unterstützt. Das Resultat der Bewertung ist zunächst rein binär (Test bestanden/nicht bestanden) ohne weitere Abstufungen. Dennoch hat sich das System bewährt und wird von studentischer Seite überwiegend angenommen.

Mit dem Sommersemester 2016 wurde jedoch eine Abstufung bei der Bewertung eingeführt, indem zu jeder Aufgabe mehrere Testfälle bereitgestellt werden. Zum Bestehen reichen nun mindestens 50 % der vorhandenen Testfälle aus. ■

## Autor

**Heiko Betz**, Wissenschaftlicher Mitarbeiter

Anschrift: Technische Universität Ilmenau, Fakultät für Informatik und Automatisierung, Fachgebiet Datenbanken und Informationssysteme, Helmholtzplatz 5, 98693 Ilmenau

E-Mail: heiko.betz@tu-ilmenau.de

## Hinweise

Die Verfasser erlauben sich, bei weiblichen und männlichen Personen die männliche oder neutrale Anrede (z.B. Teilnehmer, Mitarbeiter, Studierende/r) zu nutzen. Die nicht genannte weibliche Anredeform ist jeweils eingeschlossen.

Sämtliche Inhalte (Text, Graphik, Daten u.a.) des vorliegenden Dokuments werden im **Open Access Modus** veröffentlicht.

Sämtliche Inhalte (Text, Graphik, Daten u.a.) des vorliegenden Dokuments sind **urheberrechtlich geschützt** (© by TU Ilmenau, BASICplus, 2016-2017). Eine Nutzung ist ausschließlich im Rahmen der üblichen Zitation unter Nennung der veröffentlichten Quelle gestattet.

**Zitationsfähige Quellenangabe:** Betz, H.: *Berufsbegleitende Lehre im Fach Algorithmen und Programmierung*; BASICplus Schriftenreihe, Technische Universität Ilmenau, [www.tu-ilmenau.de/basicplus/publikationen](http://www.tu-ilmenau.de/basicplus/publikationen), 2017

**Förderhinweis:** Diese Publikation entstand im Rahmen des Projekts BASICplus „Realisierung einer offenen Studienplattform für die berufsbegleitende und durchgängige Aus- und Weiterbildung in den Ingenieurfächern“. Das Projekt wurde mit Mitteln aus dem Förderwettbewerb „Aufstieg durch Bildung: offene Hochschulen“ aus dem Programm des Bundesministeriums für Bildung und Forschung und der Länder im Rahmen der gemeinsamen Anstrengungen in der Förderung von Wissenschaft und Forschung gefördert (1. Förderphase, Laufzeit August 2014 – Januar 2018, FKZ: 16OH21017).



GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung