

Einführung in die Programmierung

Hörsaalübung 1: Motivation, Orga, Grundbegriffe

PD Dr. Thomas Hinze

Brandenburgische Technische Universität Cottbus – Senftenberg
Institut für Informatik, Informations- und Medientechnik

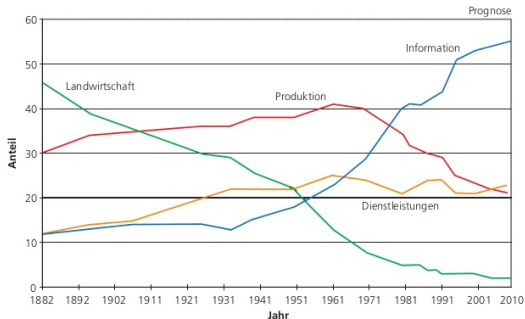
Sommersemester 2016



Brandenburgische
Technische Universität
Cottbus - Senftenberg

Informationstechnologie als riesiger Wirtschaftszweig

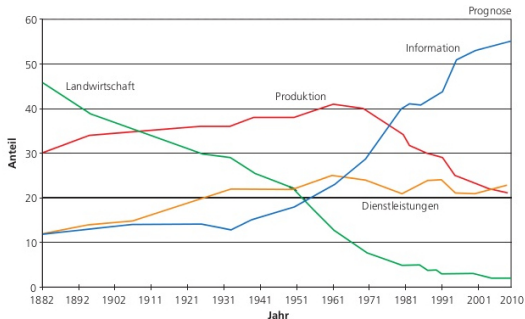
aufgewendete Arbeitszeit nach Tätigkeitsschwerpunkt (OECD, 2010)



- ca. 15% des Bruttoinlandsproduktes in Deutschland

Informationstechnologie als riesiger Wirtschaftszweig

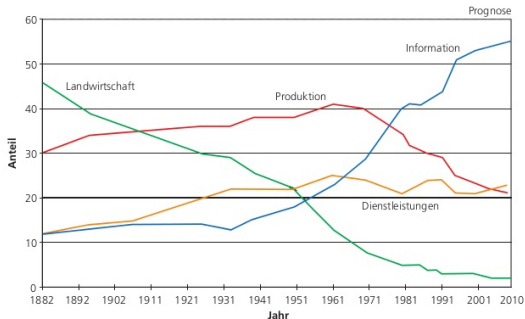
aufgewendete Arbeitszeit nach Tätigkeitsschwerpunkt (OECD, 2010)



- ca. 15% des Bruttoinlandsproduktes in Deutschland
- ca. 8 Millionen Beschäftigte in Deutschland

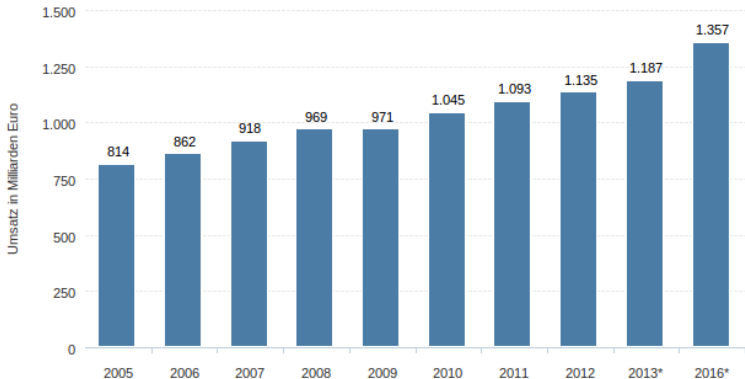
Informationstechnologie als riesiger Wirtschaftszweig

aufgewendete Arbeitszeit nach Tätigkeitsschwerpunkt (OECD, 2010)



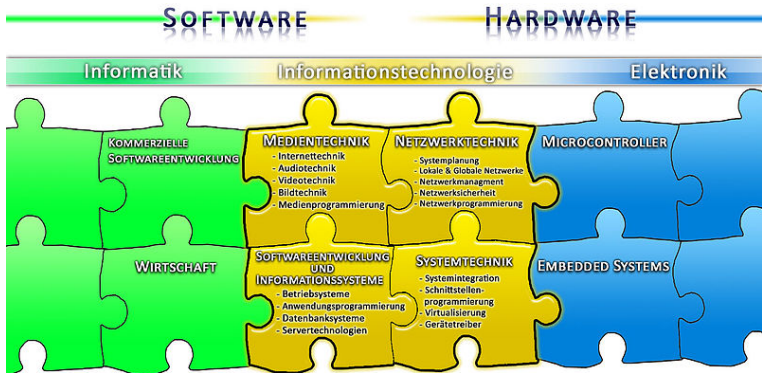
- ca. 15% des Bruttoinlandsproduktes in Deutschland
- ca. 8 Millionen Beschäftigte in Deutschland
- ca. 12% des Energieverbrauchs in Deutschland für IT

Umsatz in der gesamten IT-Branche weltweit



Quelle: IDATE

Sparten der Informationstechnologie



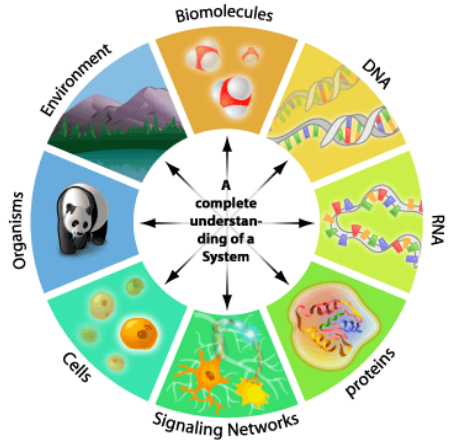
Quelle: Gesellschaft für Informatik

Programmierung essentiell in allen Sparten

Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

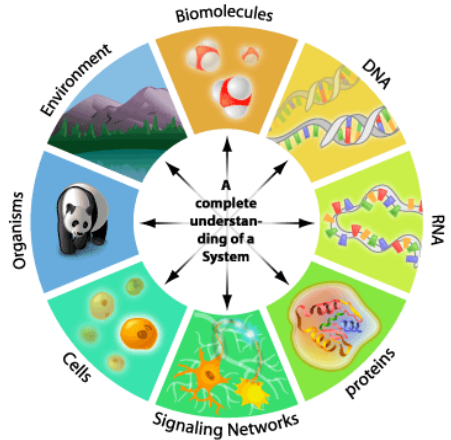
- **Bioinformatik**
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- . . .



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

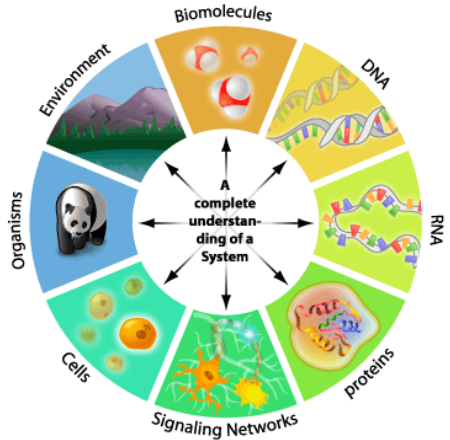
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

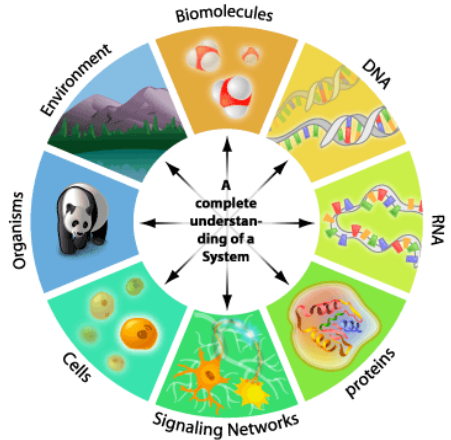
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

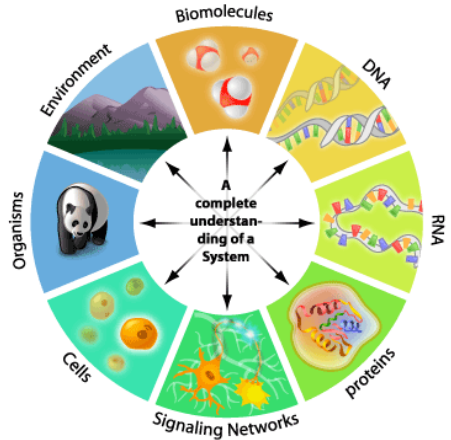
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

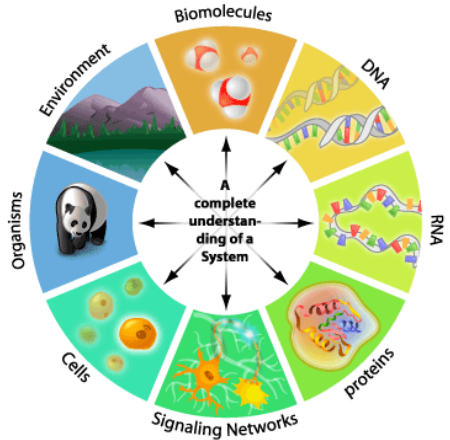
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

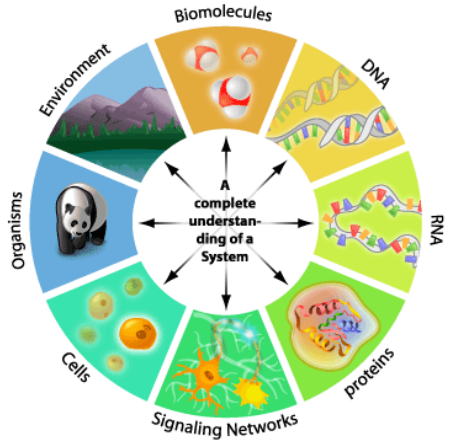
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

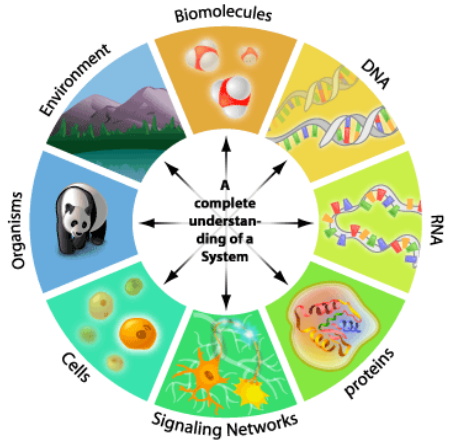
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

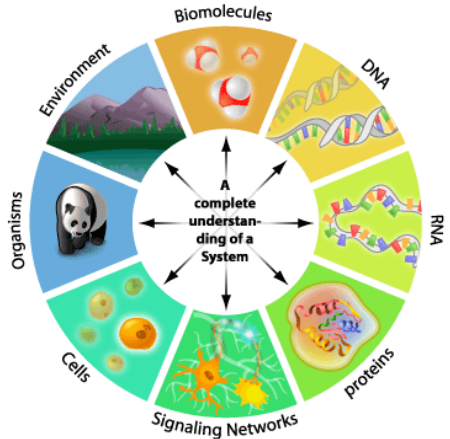
- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...



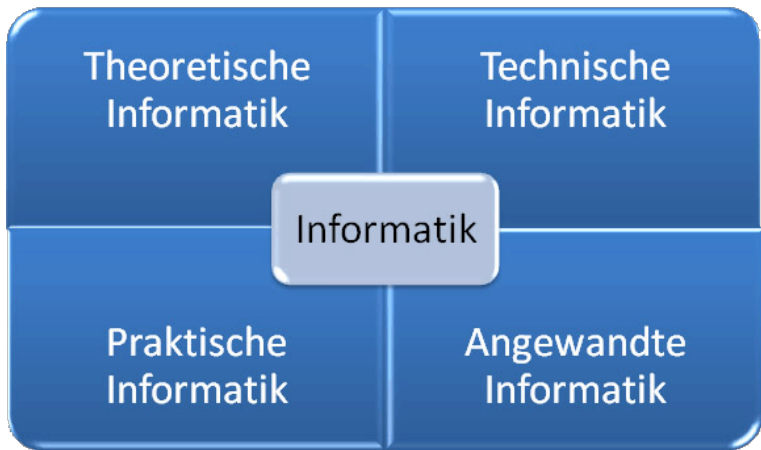
Informatik als Querschnittsdisziplin

Innovativer Dienstleister und Wegbereiter

- Bioinformatik
- Geoinformatik
- Künstliche Intelligenz
- Medieninformatik
- Medizininformatik
- Rechtsinformatik
- Verkehrsinformatik
- Wirtschaftsinformatik
- Didaktik der Informatik
- ...

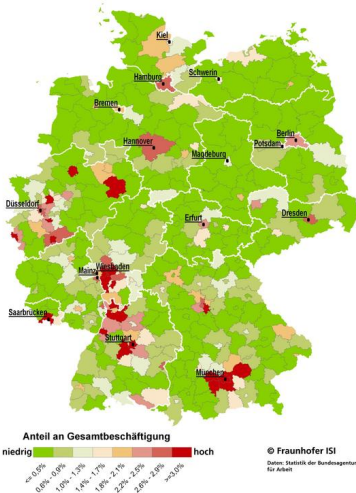


Teilgebiete der Kerninformatik



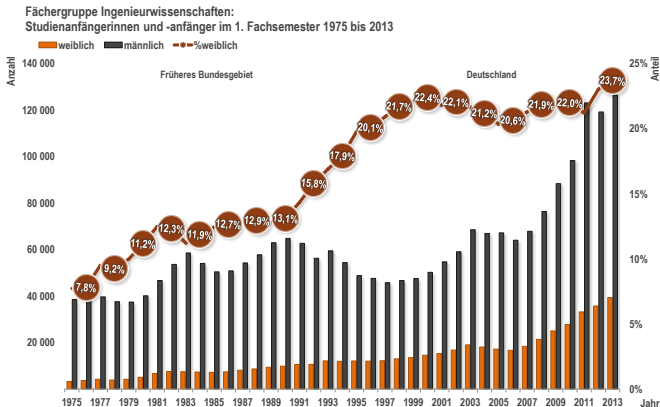
IT-Zentren in Deutschland

Anteil Software- und IT-Dienstleistungsbranche
an der Gesamtbeschäftigung 2010



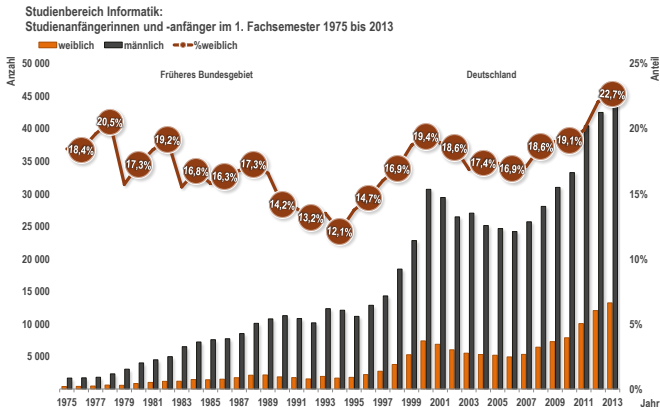
- Konzentration in den Ballungsgebieten
- Universitätsstädte und Umland
- historisch gewachsene Zentren

Studienanfängerzahlen Ingenieurwiss. in Deutschland



Quelle: Statistisches Bundesamt 2013 und eigene Berechnungen
© 2015 | Kompetenzzentrum Technik - Diversity - Chancengleichheit

Studienanfängerzahlen Informatik in Deutschland



Quelle: Statistisches Bundesamt 2013 und eigene Berechnungen
© 2015 | Kompetenzzentrum Technik - Diversity - Chancengleichheit

Erfahrung mit Softwarekunden

**Kein Kunde weiss,
was er will, aber
jeder Kunde weiss,
was er nicht will.**

aus dem Naehkaestchen der Anwendungsentwicklung

1. Motivation

2. **Organisation**

- Termine
- Informationsquellen
- Aktivitäten

3. Einordnung

Modul 12105 (6 Kreditpunkte)

Lernziele

Sie bekommen einen Einblick in die Mittel und Methoden der Softwareentwicklung und werden befähigt, einfache Programme in einer höheren Programmiersprache zu erstellen.

Inhalte

- Nutzung des PC / Laptops
- Grundlagen der Programmierung in Java: vom Problem zur Lösung, Programmiersprachen, einfache Programme, primitive Datenstrukturen, Felder und dynamische Datenstrukturen
- Funktionen: Vereinbarung und Aufruf, Parameterübergabe, Rekursion, globale und lokale Größen, Sichtbarkeit und Existenz
- Dateiarbeit: Textdateien und Binärdateien
- Algorithmen: Suchen und Sortieren, Bäume, Graphen

Modul 12105 (6 Kreditpunkte)

Lernziele

Sie bekommen einen Einblick in die Mittel und Methoden der Softwareentwicklung und werden befähigt, einfache Programme in einer höheren Programmiersprache zu erstellen.

Inhalte

- Nutzung des PC / Laptops
- Grundlagen der Programmierung in Java: vom Problem zur Lösung, Programmiersprachen, einfache Programme, primitive Datenstrukturen, Felder und dynamische Datenstrukturen
- Funktionen: Vereinbarung und Aufruf, Parameterübergabe, Rekursion, globale und lokale Größen, Sichtbarkeit und Existenz
- Dateiarbeit: Textdateien und Binärdateien
- Algorithmen: Suchen und Sortieren, Bäume, Graphen

Modul 12105 (6 Kreditpunkte)

Lernziele

Sie bekommen einen Einblick in die Mittel und Methoden der Softwareentwicklung und werden befähigt, einfache Programme in einer höheren Programmiersprache zu erstellen.

Inhalte

- Nutzung des PC / Laptops
- Grundlagen der Programmierung in Java: vom Problem zur Lösung, Programmiersprachen, einfache Programme, primitive Datenstrukturen, Felder und dynamische Datenstrukturen
- Funktionen: Vereinbarung und Aufruf, Parameterübergabe, Rekursion, globale und lokale Größen, Sichtbarkeit und Existenz
- Dateiarbeit: Textdateien und Binärdateien
- Algorithmen: Suchen und Sortieren, Bäume, Graphen

Modul 12105 (6 Kreditpunkte)

Lernziele

Sie bekommen einen Einblick in die Mittel und Methoden der Softwareentwicklung und werden befähigt, einfache Programme in einer höheren Programmiersprache zu erstellen.

Inhalte

- Nutzung des PC / Laptops
- Grundlagen der Programmierung in Java: vom Problem zur Lösung, Programmiersprachen, einfache Programme, primitive Datenstrukturen, Felder und dynamische Datenstrukturen
- Funktionen: Vereinbarung und Aufruf, Parameterübergabe, Rekursion, globale und lokale Größen, Sichtbarkeit und Existenz
- Dateiarbeit: Textdateien und Binärdateien
- Algorithmen: Suchen und Sortieren, Bäume, Graphen

Modul 12105 (6 Kreditpunkte)

Lernziele

Sie bekommen einen Einblick in die Mittel und Methoden der Softwareentwicklung und werden befähigt, einfache Programme in einer höheren Programmiersprache zu erstellen.

Inhalte

- Nutzung des PC / Laptops
- Grundlagen der Programmierung in Java: vom Problem zur Lösung, Programmiersprachen, einfache Programme, primitive Datenstrukturen, Felder und dynamische Datenstrukturen
- Funktionen: Vereinbarung und Aufruf, Parameterübergabe, Rekursion, globale und lokale Größen, Sichtbarkeit und Existenz
- Dateiarbeit: Textdateien und Binärdateien
- Algorithmen: Suchen und Sortieren, Bäume, Graphen

Modul 12105 (6 Kreditpunkte)

Lernziele

Sie bekommen einen Einblick in die Mittel und Methoden der Softwareentwicklung und werden befähigt, einfache Programme in einer höheren Programmiersprache zu erstellen.

Inhalte

- Nutzung des PC / Laptops
- Grundlagen der Programmierung in Java: vom Problem zur Lösung, Programmiersprachen, einfache Programme, primitive Datenstrukturen, Felder und dynamische Datenstrukturen
- Funktionen: Vereinbarung und Aufruf, Parameterübergabe, Rekursion, globale und lokale Größen, Sichtbarkeit und Existenz
- Dateiarbeit: Textdateien und Binärdateien
- Algorithmen: Suchen und Sortieren, Bäume, Graphen

Einordnung in das Studium

- Bauingenieurwesen – B.Sc. PO 2011
- Betriebswirtschaftslehre – B.Sc. PO 2011
- Elektrotechnik – B.Sc. PO 2006
- Maschinenbau – B.Sc. PO 2006
- Verfahrenstechnik – B.Sc. PO 2005
- Wirtschaftsingenieurwesen – B.Sc. PO 2008
- weitere 31 Studiengänge

Termine

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
07:30 – 09:00			Laboruebung VG1C/0.03 Thomas Hinze		
09:15 – 10:45					Selbsttrainingslot VG1C/0.03
11:30 – 13:00	Selbsttrainingslot VG1C/0.03		Laboruebung VG1C/0.03 Martin Buschack		Selbsttrainingslot VG1C/0.03
13:45 – 15:15			Hoersaaluebung HS 2 Thomas Hinze		Selbsttrainingslot VG1C/0.03
15:30 – 17:00		Vorlesung Audimax 2 Thomas Hinze	Laboruebung VG1C/0.03 Benjamin Foerster	Laboruebung VG1C/0.03 Thomas Hinze	
17:30 – 19:00		Laboruebung VG1C/0.03 Thomas Hinze	Hoersaaluebung HG 0.20 Thomas Hinze		

- Vorlesung, Hörsaalübung und Laborübung wöchentlich
- Vorlesungszeitraum: 04.04.-15.07.2016

Webseite

<http://www.informatik.tu-cottbus.de/~hinzet/eidp-sose16/>

Einführung in die Programmierung

Sommersemester 2016 an der BTU Cottbus - Senftenberg

Modul 12105, 6 Kreditpunkte

wöchentlich 2 SWS Vorlesung, 2 SWS Hörsaalübung und 2 SWS Laborübung

Zielgruppe: verschiedene Ingenieurstudiengänge sowie fachübergreifendes Studium

Aktuelles

Übungsgruppen

Aufgabenblätter und Material

Klausuren

Modulbeschreibung

moodle

News

Einschreibung
für Laborübung im
moodle ab
04.04.2016
freigeschaltet

Aktuelles

Herzlich willkommen! Wir freuen uns, Sie zur Veranstaltung *Einführung in die Programmierung* zu begrüßen. Programmieren ist kein Buch mit sieben Siegeln, sondern ein leicht erlernbares Handwerk, das in Studium und Beruf immer wieder nützlich ist und sogar Spaß bereiten kann. Lassen Sie uns gemeinsam die ersten Schritte in der Programmierung gehen. Als Programmiersprache wählen wir das weitverbreitete und bewährte Java. Anhand kleiner Beispiele und anschaulicher Aufgaben erschließen wir uns diese Programmiersprache und trainieren, wie man damit erfolgreich kleine Programme schreibt und zum Laufen bringt.

- Ankündigungen (z.B. Klausurtermine)
- Übungsblätter zum Download und hilfreiche Links
- Material zum Download

Modulanmeldung

<https://www.zv.tu-cottbus.de/qispos11/rds?state=user&type=0>

Wichtig, damit Ihre Note vom Prüfungsamt verbucht wird.

- Modulanmeldung über das *online-Portal*
- Bitte innerhalb der ersten drei Vorlesungswochen des Semesters, also spätestens bis *22.04.2016*
- bei Bedarf Rücktritt von der Modulanmeldung bis Ende der siebten Vorlesungswoche ebenfalls über online-Portal möglich, also bis 20.05.2016
- Die Modulanmeldung bitte zusätzlich zur Einschreibung im moodle vornehmen

Modulanmeldung

<https://www.zv.tu-cottbus.de/qispos11/rds?state=user&type=0>

Wichtig, damit Ihre Note vom Prüfungsamt verbucht wird.

- Modulanmeldung über das *online-Portal*
- Bitte innerhalb der ersten drei Vorlesungswochen des Semesters, also spätestens bis *22.04.2016*
- bei Bedarf Rücktritt von der Modulanmeldung bis Ende der siebten Vorlesungswoche ebenfalls über online-Portal möglich, also bis 20.05.2016
- Die Modulanmeldung bitte zusätzlich zur Einschreibung im moodle vornehmen

Modulanmeldung

<https://www.zv.tu-cottbus.de/qispos11/rds?state=user&type=0>

Wichtig, damit Ihre Note vom Prüfungsamt verbucht wird.

- Modulanmeldung über das *online-Portal*
- Bitte innerhalb der ersten drei Vorlesungswochen des Semesters, also spätestens bis *22.04.2016*
- bei Bedarf Rücktritt von der Modulanmeldung bis Ende der siebten Vorlesungswoche ebenfalls über online-Portal möglich, also bis 20.05.2016
- Die Modulanmeldung bitte zusätzlich zur Einschreibung im moodle vornehmen

Modulanmeldung

<https://www.zv.tu-cottbus.de/qispos11/rds?state=user&type=0>

Wichtig, damit Ihre Note vom Prüfungsamt verbucht wird.

- Modulanmeldung über das *online-Portal*
- Bitte innerhalb der ersten drei Vorlesungswochen des Semesters, also spätestens bis *22.04.2016*
- bei Bedarf Rücktritt von der Modulanmeldung bis Ende der siebten Vorlesungswoche ebenfalls über online-Portal möglich, also bis 20.05.2016
- Die Modulanmeldung bitte zusätzlich zur Einschreibung im moodle vornehmen

Zu erbringende Prüfungsleistungen

- erfolgreiche Bearbeitung der Übungsaufgaben
(Abgabe regelmäßig etwa alle 14 Tage über moodle)

Zu erbringende Prüfungsleistungen

- erfolgreiche Bearbeitung der Übungsaufgaben (Abgabe regelmäßig etwa alle 14 Tage über moodle)
- **Zwischenklausur** am Di, **10.05.2016**, 15:45-16:45 Uhr anstelle der Vorlesung

Zu erbringende Prüfungsleistungen

- erfolgreiche Bearbeitung der Übungsaufgaben
(Abgabe regelmäßig etwa alle 14 Tage über moodle)
- **Zwischenklausur** am Di, **10.05.2016**, 15:45-16:45 Uhr
anstelle der Vorlesung
- **Endklausur** am Mo, **01.08.2016**, 08:00-10:00 Uhr, Großer
Hörsaal

Zu erbringende Prüfungsleistungen

- erfolgreiche Bearbeitung der Übungsaufgaben
(Abgabe regelmäßig etwa alle 14 Tage über moodle)
- **Zwischenklausur** am Di, **10.05.2016**, 15:45-16:45 Uhr
anstelle der Vorlesung
- **Endklausur** am Mo, **01.08.2016**, 08:00-10:00 Uhr, Großer
Hörsaal
- **Nachschreibe-/Wiederholungstermin:**
Di, **20.09.2015**, 13:00-16:00 Uhr, HS 2

Zu erbringende Prüfungsleistungen

- erfolgreiche Bearbeitung der Übungsaufgaben (Abgabe regelmäßig etwa alle 14 Tage über moodle)
- **Zwischenklausur** am Di, **10.05.2016**, 15:45-16:45 Uhr anstelle der Vorlesung
- **Endklausur** am Mo, **01.08.2016**, 08:00-10:00 Uhr, Großer Hörsaal
- **Nachschreibe-/Wiederholungstermin:**
Di, **20.09.2015**, 13:00-16:00 Uhr, HS 2

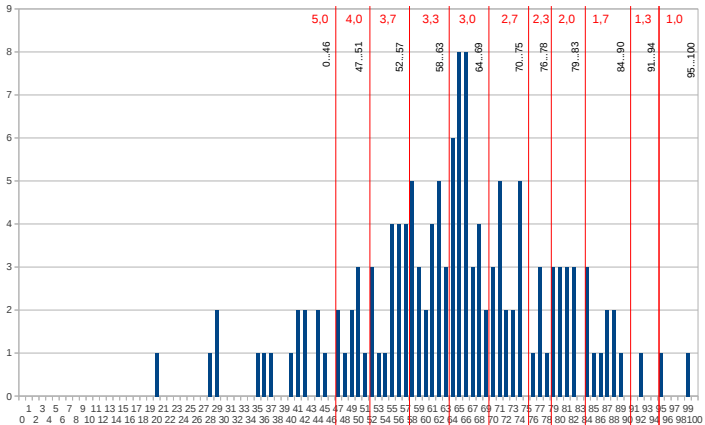
„Das Modul ist bestanden, wenn die Übungsblätter erfolgreich bearbeitet sind und die Gesamtnote aus beiden Klausuren mindestens ausreichend ist. Die Gesamtnote des Moduls ergibt sich zu 30% aus der Zwischenklausur und zu 70% aus der Abschlussklausur.“

Punkteverteilung und Noten im WiSe 2012/13

$$\text{Prozente} = 30 \cdot \frac{\text{erreichte Punktzahl Zwischenklausur}}{\text{erreichbare Punktzahl Zwischenklausur}} + 70 \cdot \frac{\text{erreichte Punktzahl Endklausur}}{\text{erreichbare Punktzahl Endklausur}}$$

Verteilung prozentuale Leistungsstände EidP Klausuren 23.11.12 und 25.02.13

$$30 * \text{Punktzahl_Zwischenklausur} / 60 + 70 * \text{Punktzahl_Endklausur} / 100$$



Abgabe der Übungslösungen

- insgesamt 6 Übungsblätter, zu finden auf Übungswebseite
- davon bitte mindestens 5 erfolgreich lösen

Abgabe der Übungslösungen

- insgesamt 6 Übungsblätter, zu finden auf Übungswebseite
- davon bitte mindestens 5 erfolgreich lösen
 - Bitte schriftliche Lösung zu den einzelnen Aufgaben erstellen und als **pdf** abspeichern (notfalls auch leserlich handgeschriebene Blätter an öffentlichem Uni-Kopierer als pdf auf USB-Stick kostenfrei einscannen)

Abgabe der Übungslösungen

- insgesamt 6 Übungsblätter, zu finden auf Übungswebseite
- davon bitte mindestens 5 erfolgreich lösen
 - Bitte schriftliche Lösung zu den einzelnen Aufgaben erstellen und als **pdf** abspeichern (notfalls auch leserlich handgeschriebene Blätter an öffentlichem Uni-Kopierer als pdf auf USB-Stick kostenfrei einscannen)
 - Wenn laut Übungsblatt gefordert, bitte zusätzlich Programmquelltexte als `.java`-Dateien erstellen

Abgabe der Übungslösungen

- insgesamt 6 Übungsblätter, zu finden auf Übungswebseite
- davon bitte mindestens 5 erfolgreich lösen
 - Bitte schriftliche Lösung zu den einzelnen Aufgaben erstellen und als **pdf** abspeichern (notfalls auch leserlich handgeschriebene Blätter an öffentlichem Uni-Kopierer als pdf auf USB-Stick kostenfrei einscannen)
 - Wenn laut Übungsblatt gefordert, bitte zusätzlich Programmquelltexte als `.java`-Dateien erstellen
 - Bitte pdf und alle `.java`-Dateien eines Übungsblattes in ein **zip**-Archiv packen

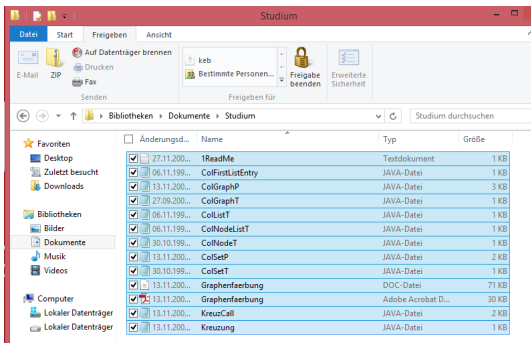
Abgabe der Übungslösungen

- insgesamt 6 Übungsblätter, zu finden auf Übungswebseite
- davon bitte mindestens 5 erfolgreich lösen
 - Bitte schriftliche Lösung zu den einzelnen Aufgaben erstellen und als **pdf** abspeichern (notfalls auch leserlich handgeschriebene Blätter an öffentlichem Uni-Kopierer als pdf auf USB-Stick kostenfrei einscannen)
 - Wenn laut Übungsblatt gefordert, bitte zusätzlich Programmquelltexte als `.java`-Dateien erstellen
 - Bitte pdf und alle `.java`-Dateien eines Übungsblattes in ein **zip**-Archiv packen
 - zip-Archiv bitte rechtzeitig im moodle hochladen (bis Abgabetag 23:55 Uhr MESZ freigeschaltet)

Abgabe der Übungslösungen

- insgesamt 6 Übungsblätter, zu finden auf Übungswebseite
- davon bitte mindestens 5 erfolgreich lösen
 - Bitte schriftliche Lösung zu den einzelnen Aufgaben erstellen und als **pdf** abspeichern (notfalls auch leserlich handgeschriebene Blätter an öffentlichem Uni-Kopierer als pdf auf USB-Stick kostenfrei einscannen)
 - Wenn laut Übungsblatt gefordert, bitte zusätzlich Programmquelltexte als `.java`-Dateien erstellen
 - Bitte pdf und alle `.java`-Dateien eines Übungsblattes in ein **zip**-Archiv packen
 - zip-Archiv bitte rechtzeitig im moodle hochladen (bis Abgabetag 23:55 Uhr MESZ freigeschaltet)
- Lösungsideen und -ansätze können gemeinsam in Teams erarbeitet werden, aber jeder gibt bitte seine individuelle Lösung im moodle ab. Bitte keine Plagiate einreichen.

Dateien in Zip-Archiv packen und entpacken



unter Windows (XP, 7, 8, 10)

- Dateien und/oder Ordner auswählen
- Menüpunkt „Freigeben“ anklicken, auf Icon ZIP klicken
- Archivdatei wird wie ein Ordner behandelt

Zip-Archivformat ist plattformübergreifend, deshalb bevorzugt

Bonuspunkte und Programmierwettbewerb

- Jeder Teilnehmer kann **bis zu 5 Bonuspunkte** erwirtschaften.
- Bonuspunkte zu Klausurpunkten hinzuaddiert
- 5 Punkte entspricht etwa einer Notenstufe (z.B. von 1,7 auf 1,3)

Bonuspunkte und Programmierwettbewerb

- Jeder Teilnehmer kann **bis zu 5 Bonuspunkte** erwirtschaften.
- Bonuspunkte zu Klausurpunkten hinzuaddiert
- 5 Punkte entspricht etwa einer Notenstufe (z.B. von 1,7 auf 1,3)

Alle 6 Übungsblätter termingemäß erfolgreich gelöst:
2 Bonuspunkte

Bonuspunkte und Programmierwettbewerb

- Jeder Teilnehmer kann **bis zu 5 Bonuspunkte** erwirtschaften.
- Bonuspunkte zu Klausurpunkten hinzuaddiert
- 5 Punkte entspricht etwa einer Notenstufe (z.B. von 1,7 auf 1,3)

Alle 6 Übungsblätter termingemäß erfolgreich gelöst:
2 Bonuspunkte

Programmierwettbewerb

- Im Juni 2016 Aufgabe(n) zusätzlich zu den Übungsblättern

Bonuspunkte und Programmierwettbewerb

- Jeder Teilnehmer kann **bis zu 5 Bonuspunkte** erwirtschaften.
- Bonuspunkte zu Klausurpunkten hinzuaddiert
- 5 Punkte entspricht etwa einer Notenstufe (z.B. von 1,7 auf 1,3)

Alle 6 Übungsblätter termingemäß erfolgreich gelöst:
2 Bonuspunkte

Programmierwettbewerb

- Im Juni 2016 Aufgabe(n) zusätzlich zu den Übungsblättern
- Programmierlösung selbstständig erarbeiten, zum Stichtag im moodle hochladen

Bonuspunkte und Programmierwettbewerb

- Jeder Teilnehmer kann **bis zu 5 Bonuspunkte** erwirtschaften.
- Bonuspunkte zu Klausurpunkten hinzuaddiert
- 5 Punkte entspricht etwa einer Notenstufe (z.B. von 1,7 auf 1,3)

Alle 6 Übungsblätter termingemäß erfolgreich gelöst:
2 Bonuspunkte

Programmierwettbewerb

- Im Juni 2016 Aufgabe(n) zusätzlich zu den Übungsblättern
- Programmierlösung selbstständig erarbeiten, zum Stichtag im moodle hochladen
- Wer **zuerst** einen eigenständigen Quelltext einreicht, der die Aufgabe **vollständig erfüllt**, erhält **5 Bonuspunkte**, der nächste **4**, der nächste **3** und je **2** für alle weiteren fristgemäßen, korrekten und individuell erarbeiteten Einreichungen

Bonuspunkte und Programmierwettbewerb

- Jeder Teilnehmer kann **bis zu 5 Bonuspunkte** erwirtschaften.
- Bonuspunkte zu Klausurpunkten hinzuaddiert
- 5 Punkte entspricht etwa einer Notenstufe (z.B. von 1,7 auf 1,3)

Alle 6 Übungsblätter termingemäß erfolgreich gelöst:
2 Bonuspunkte

Programmierwettbewerb

- Im Juni 2016 Aufgabe(n) zusätzlich zu den Übungsblättern
- Programmierlösung selbstständig erarbeiten, zum Stichtag im moodle hochladen
- Wer **zuerst** einen eigenständigen Quelltext einreicht, der die Aufgabe **vollständig erfüllt**, erhält **5 Bonuspunkte**, der nächste **4**, der nächste **3** und je **2** für alle weiteren fristgemäßen, korrekten und individuell erarbeiteten Einreichungen

⇒ **Mehr als 5 Bonuspunkte dürfen pro Person nicht vergeben werden, Bonuspunkte sind nicht übertragbar.**

Anerkennung älterer Übungslösungen

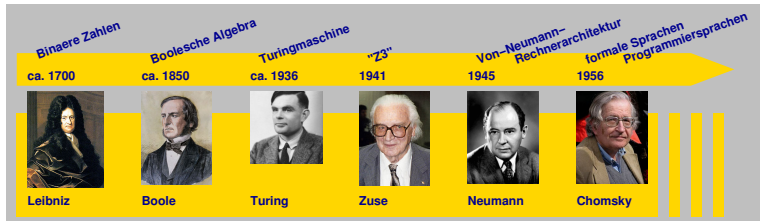
Bereits bewertete Übungslösungen aus dem **SoSe 15** und aus dem **WiSe 15/16** können im SoSe16 anerkannt werden. Bitte schreiben Sie dazu eine E-Mail an mich (`thomas.hinze@b-tu.de`).

Zwischenklausurpunkte, Endklausurpunkte und Bonuspunkte aus zurückliegenden Semestern dürfen leider nicht ins laufende Semester übertragen werden.

1. Motivation
2. Organisation
3. **Einordnung**
 - Begriffe
 - Geschichte
 - Geschichten

Informatik

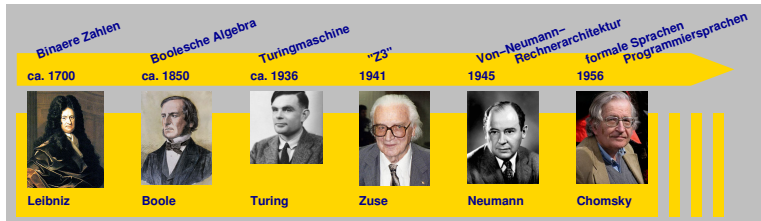
Informatik ist die Wissenschaft von der systematischen und automatischen Verarbeitung von Informationen.



Informatik

Informatik ist die Wissenschaft von der systematischen und automatischen Verarbeitung von Informationen.

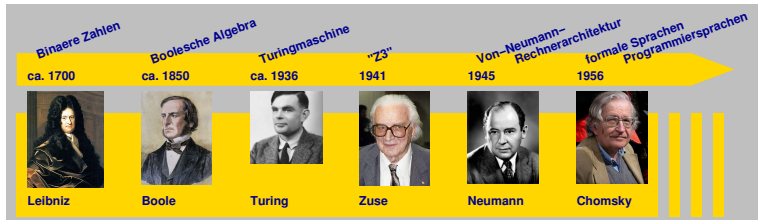
- Wurzeln in der Mathematik: Formalisierung und Automatisierung von Berechnungen



Informatik

Informatik ist die Wissenschaft von der systematischen und automatischen Verarbeitung von Informationen.

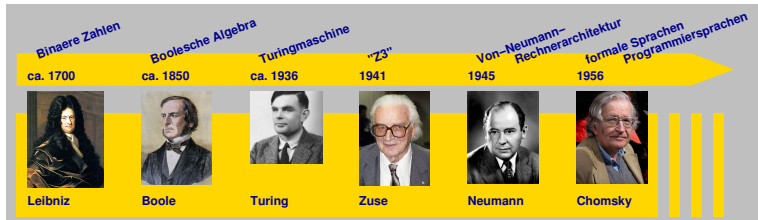
- Wurzeln in der Mathematik: Formalisierung und Automatisierung von Berechnungen
- Begriff als Verschmelzung von „Information“ und „Mathematik“, erstmalig erwähnt 1957



Informatik

Informatik ist die Wissenschaft von der systematischen und automatischen Verarbeitung von Informationen.

- Wurzeln in der Mathematik: Formalisierung und Automatisierung von Berechnungen
- Begriff als Verschmelzung von „Information“ und „Mathematik“, erstmalig erwähnt 1957
- Seit 1969 Informatikstudium in Deutschland



Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Algorithmus

**Ein Algorithmus ist eine aus endlich vielen Schritten bestehende eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
(intuitiver Algorithmusbegriff)**

- *Fintheit* (endliche Länge der Beschreibung)
- *Ausführbarkeit* und *Effektivität*
- *dynamische Fintheit* (zu jedem Zeitpunkt endlich viele Ressourcen wie Speicherplatz)
- *Terminierung* (Ausführung endet nach endlich vielen Schritten)

Oft zusätzlich gefordert:

- *Determiniertheit* (unter gleichen Voraussetzungen stets gleiches Ergebnis)
- *Determinismus* (nächster anzuwendender Verfahrensschritt eindeutig definiert)



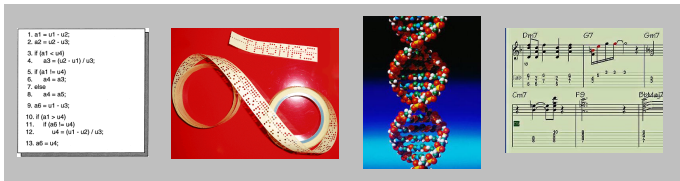
Al-Chwarizmi
um 825

Arabische Begriffe
Algorithmus
Algebra
Alkohol
Alchemie

Programm

Ein (Computer)Programm ist eine Folge von den Regeln der jeweiligen Programmiersprache genügenden Anweisungen, die auf einem Computer ausgeführt werden können, um damit eine bestimmte Funktionalität zur Verfügung zu stellen.

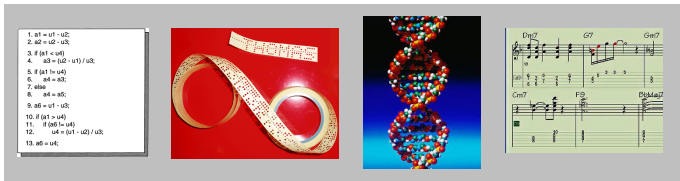
- Programm: griechisch „Vorschrift“
- Programm als Implementierung (Umsetzung) eines oder mehrerer Algorithmen in einer Programmiersprache
- Software: Sammelbegriff für ausführbare Programme und zugehörige Daten



Programm

Ein (Computer)Programm ist eine Folge von den Regeln der jeweiligen Programmiersprache genügenden Anweisungen, die auf einem Computer ausgeführt werden können, um damit eine bestimmte Funktionalität zur Verfügung zu stellen.

- Programm: griechisch „Vorschrift“
- Programm als Implementierung (Umsetzung) eines oder mehrerer Algorithmen in einer Programmiersprache
- Software: Sammelbegriff für ausführbare Programme und zugehörige Daten



```

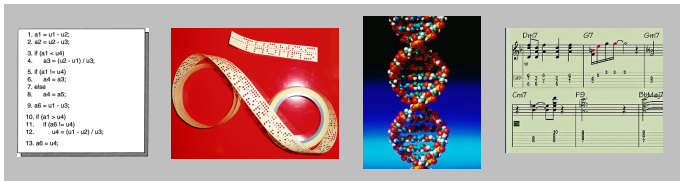
1. a1 = u1 - u2;
2. a2 = u2 - u3;
3. if (a1 < u4)
4.   u2 = (u2 - u1) / u3;
5. if (a1 = u4)
6.   a4 = a2;
7. else
8.   a4 = a5;
9. a5 = u1 - u3;
10. if (a1 = u4)
11.   if (a5 = u4)
12.     u4 = (u1 - u2) / u3;
13. a5 = u4;
    
```



Programm

Ein (Computer)Programm ist eine Folge von den Regeln der jeweiligen Programmiersprache genügenden Anweisungen, die auf einem Computer ausgeführt werden können, um damit eine bestimmte Funktionalität zur Verfügung zu stellen.

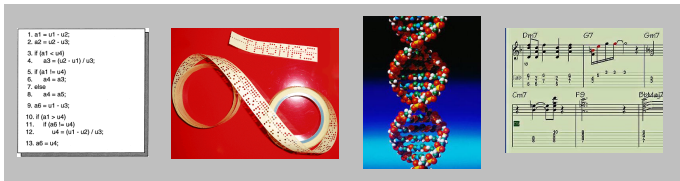
- Programm: griechisch „Vorschrift“
- Programm als Implementierung (Umsetzung) eines oder mehrerer Algorithmen in einer Programmiersprache
- Software: Sammelbegriff für ausführbare Programme und zugehörige Daten



Programm

Ein (Computer)Programm ist eine Folge von den Regeln der jeweiligen Programmiersprache genügenden Anweisungen, die auf einem Computer ausgeführt werden können, um damit eine bestimmte Funktionalität zur Verfügung zu stellen.

- Programm: griechisch „Vorschrift“
- Programm als Implementierung (Umsetzung) eines oder mehrerer Algorithmen in einer Programmiersprache
- Software: Sammelbegriff für ausführbare Programme und zugehörige Daten



Programmierung und Programmiersprachen

Programmierung

Zweck: Modellierung und Lösung eines Problems

- Design und Entwicklung von *Algorithmen* und
- geeigneten *Datenstrukturen*
(Datentyp + Operationen, die auf ihm unterstützt werden)
- *Implementierung* in einer Programmiersprache

Ziele:

- Korrektheit und Robustheit
- Effizienz (Zeit- und Platzbedarf)
- Lesbarkeit, Wartbarkeit und Testbarkeit

Programmierung und Programmiersprachen

Programmierung

Zweck: Modellierung und Lösung eines Problems

- Design und Entwicklung von *Algorithmen* und
- geeigneten *Datenstrukturen*
(Datentyp + Operationen, die auf ihm unterstützt werden)
- *Implementierung* in einer Programmiersprache

Ziele:

- Korrektheit und Robustheit
- Effizienz (Zeit- und Platzbedarf)
- Lesbarkeit, Wartbarkeit und Testbarkeit

Programmierung und Programmiersprachen

Programmierung

Zweck: Modellierung und Lösung eines Problems

- Design und Entwicklung von *Algorithmen* und
- geeigneten *Datenstrukturen*
(Datentyp + Operationen, die auf ihm unterstützt werden)
- *Implementierung* in einer Programmiersprache

Ziele:

- Korrektheit und Robustheit
- Effizienz (Zeit- und Platzbedarf)
- Lesbarkeit, Wartbarkeit und Testbarkeit

Programmierung und Programmiersprachen

Programmierung

Zweck: Modellierung und Lösung eines Problems

- Design und Entwicklung von *Algorithmen* und
- geeigneten *Datenstrukturen*
(Datentyp + Operationen, die auf ihm unterstützt werden)
- *Implementierung* in einer Programmiersprache

Ziele:

- Korrektheit und Robustheit
- Effizienz (Zeit- und Platzbedarf)
- Lesbarkeit, Wartbarkeit und Testbarkeit

Programmierung und Programmiersprachen

Programmierung

Zweck: Modellierung und Lösung eines Problems

- Design und Entwicklung von *Algorithmen* und
- geeigneten *Datenstrukturen*
(Datentyp + Operationen, die auf ihm unterstützt werden)
- *Implementierung* in einer Programmiersprache

Ziele:

- Korrektheit und Robustheit
- Effizienz (Zeit- und Platzbedarf)
- Lesbarkeit, Wartbarkeit und Testbarkeit