

Algorithmen und Datenstrukturen

- Organisation, Einführung und Übersicht -

Prof. Dr. Klaus Volbert

Wintersemester 2018/19
Regensburg, 01. Oktober 2018

Algorithmen in der Informatik

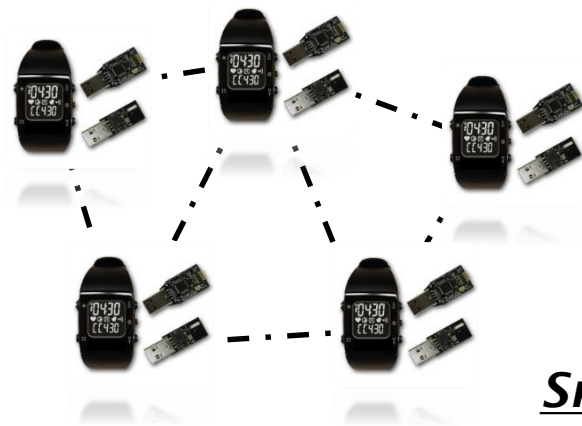
- Welche **Programmiersprachen** beherrschen Sie?
- Welche **Konzepte** kennen Sie aus Programmiersprachen?
- Was wissen Sie über **Algorithmen**?
- Welche Algorithmen kennen Sie?
- Welche **Datenstrukturen** kennen Sie?
- Was erwarten **Sie** von dieser Veranstaltung?

- Studiengang Bachelor Informatik (IN3/IN4)
 - Vorlesungen:
 - Mo 13:30-15:00 Uhr Raum K001
 - Do 08:15-09:45 Uhr Raum U212
 - Übungen (**Beginn ab Donnerstag, 04.10.2018**):
 - Gruppe 1: Di 08:15-09:45 Uhr Raum K223 (IN3IN4[1])
 - Gruppe 2: Di 11:45-13:15 Uhr Raum K140 (IN3IN4[2])
 - Gruppe 3: Mo 15:15-16:45 Uhr Raum K220 (IN3IN4[3])
 - Gruppe 4: Do 10:00-11:30 Uhr Raum K140 (IN3IN4[4])
- Rahmendaten zur Veranstaltung
 - Nach Studienplan: 3./4. Semester
 - Mindestens 30 Credits aus dem 1. Studienabschnitt
 - 6 SWS: V4, Ü2, 8 Credits (nach erfolgreich bestandener Prüfung)

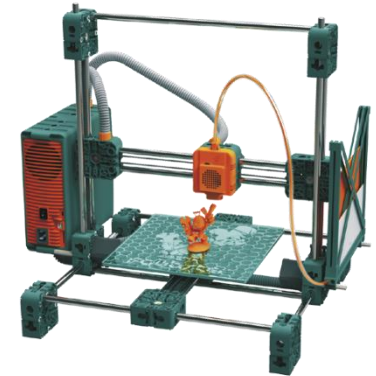
- Begleitende Webseite mit aktuellen Informationen
 - <http://hps.oth-regensburg.de/~vok39696>
 - Lehre Wintersemester 2018/19:
 - Algorithmen und Datenstrukturen IN (→ [Kursraum in G.R.I.P.S.](#))
- Dozent
 - Sprechstunde: Di 10-11 Uhr Raum K214 und nach Vereinbarung
`klaus.volbert@oth-regensburg.de`
 - Lehrgebiete:
 - Algorithmen und Datenstrukturen, Spezielle Algorithmen
 - Modelle und Algorithmen im Internet der Dinge, ...

Forschungsinteressen

Algorithmen-/Software-/App-Entwicklung für drahtlose, eingebettete Systeme (z. B. **Mobile Ad-hoc-Netze**, **Sensornetze**, **IoT**, **Industrie 4.0**)



Algorithmen
für **3D-Drucker**

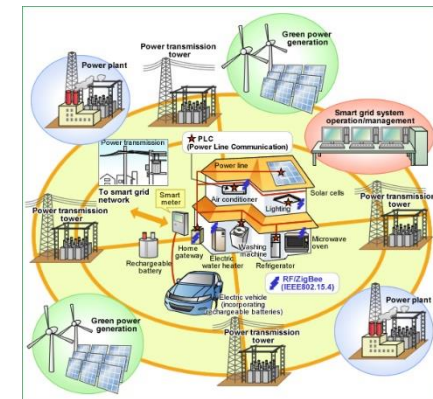


Smarte Algorithmen

OTH Forschungscluster **IKT** und **RAKS**



- Studentische Hilfskräfte
- Themen für Abschlussarbeiten
- Themen für HSP im Master



Energieinformatik (**RCER**)
(**Smart Metering**, **Smart Submetering**, **Smart Grid**)

Spielregeln in der Vorlesung

- Seminaristischer Unterricht
 - Aktive Teilnahme erwünscht
- Verwendung elektronischer Geräte
 - Die **kontextunabhängige** Verwendung von Tablets, Laptops, Mobiltelefonen und ähnlichen elektronischen Geräten während der Vorlesung ist **nicht** zugelassen



Bitte ausschalten!

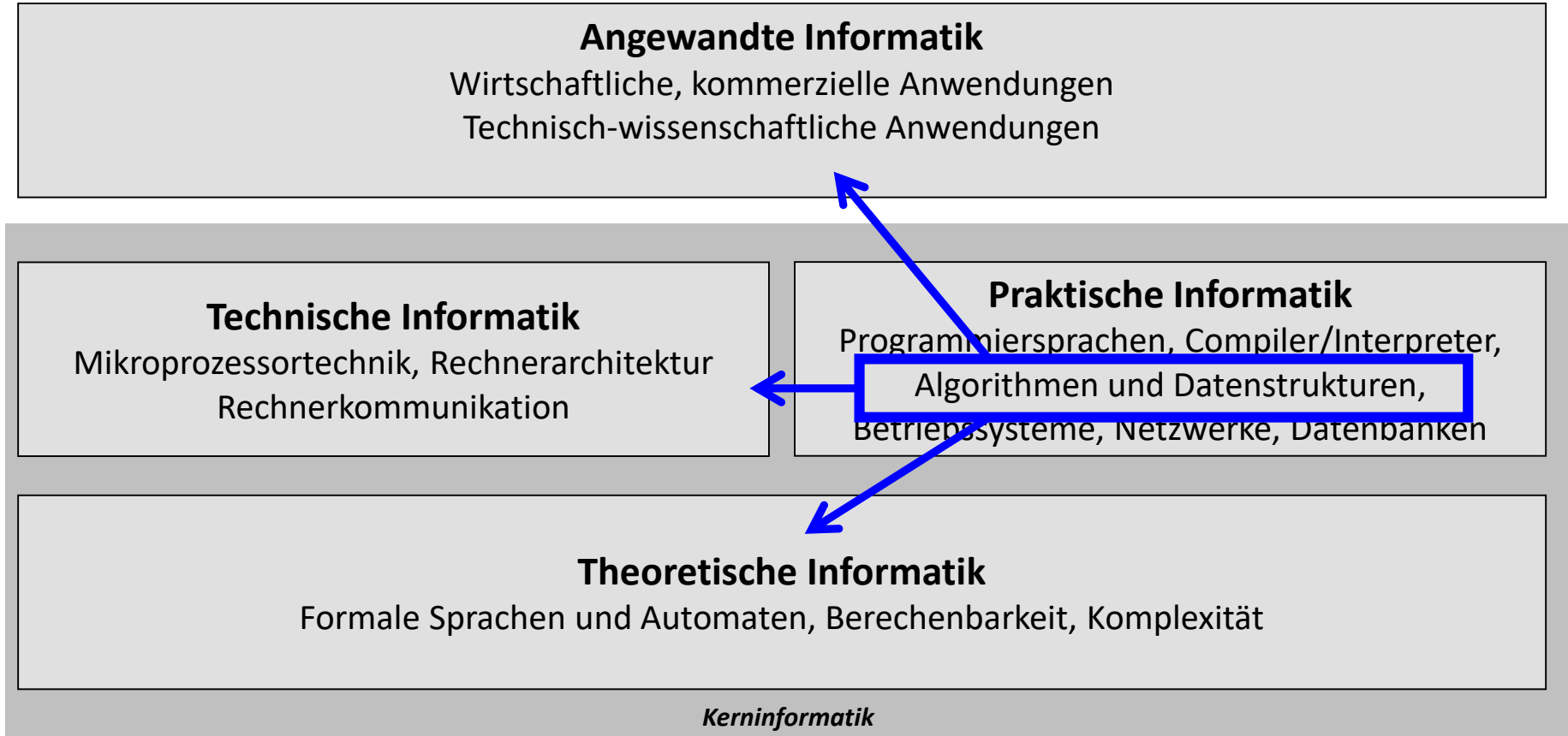
• Übungsbetrieb

- Übungsgruppenauswahl im Kursraum (Zuordnung randomisiert)
- Jede Woche am **Mittwoch**: 1 Übungsblatt mit 4 Aufgaben
- Bearbeitung der Aufgaben
 - Vorbereitung/Bearbeitung frühzeitig starten
 - Bearbeitung in den Übungen fortsetzen (Dozent kann unterstützen)
 - Fertigstellung nach den Übungen (**nacharbeiten und üben!!!**)
 - Spätestens bis zum folgenden **Dienstag (23:00 Uhr)**:
 - Votieren der Aufgaben und elektronische Abgabe Ihrer Lösungen durch Hochladen einer Datei (**Pflichtabgabe**)
- Nutzen Sie die Chance zum **Fragen** und zum **Präsentieren** Ihrer Lösung!

- Art der Prüfung
 - 90 min. Klausur am Ende des Semesters
- Zugelassene Hilfsmittel
 - Ein beidseitig handgeschriebenes DIN-A4 Blatt
 - Taschenrechner oder andere elektronische Hilfsmittel sind **nicht** erlaubt
- Relevante Themen
 - Gesamter Stoff der Veranstaltung
 - Klausuraufgaben angelehnt an Übungsaufgaben
- Zulassungsvoraussetzung
 - Regelmäßige und aktive Teilnahme an den Übungen
 - **50 % der Aufgaben positiv votiert und abgegeben**

- Spätestens vor Erscheinen des neuen Übungsblatts (bis Di 23 Uhr)
...Aufgabe votieren und Lösungen elektronisch abgeben!
- Votieren
 - Zu jeder Aufgabe muss jeder Teilnehmer angeben, ob die Aufgabe gelöst werden konnte
 - 50 % der Aufgaben müssen gelöst werden (**positiv votiert werden**)
- Abgabe
 - Ihre Lösungen zu positiv votierten Aufgaben müssen elektronisch durch Hochladen einer Datei abgegeben werden
 - Die Datei kann ein Zip-Archiv oder eine PDF-/JPG-Datei sein
- Wichtig:
 - Nehmen Sie sich genügend Zeit zur Bearbeitung der Aufgaben!
 - Eine Aufgabe, die schwierig wirkt, kann auch einfach zu lösen sein
 - Lösungen müssen nicht perfekt sein

Einordnung der Veranstaltung



Anmerkung: Informatik \neq Programmierung

Zentraler Begriff: Algorithmus

- Persischer Mathematiker und Astronom (~ 825 n. Chr.)
Abu Ja'far Muhammad Ibn Musa al-Khwarizmi



„Kitab al jabr w'al mugabala“

- Intuitive, informale Definition eines Algorithmus:
 - Ein Algorithmus ist eine Vorschrift zur Lösung eines Problems, die für eine Realisierung in Form eines Programms auf einem Computer geeignet ist (Taschenbuch der Informatik, 2004)
 - Ein Algorithmus ist eine präzise Handlungsvorschrift, um aus vorgegebenen **Eingaben** in endlich vielen Schritten eine bestimmte **Ausgabe** zu ermitteln (Eingabe-Verarbeitung-Ausgabe, EVA-Prinzip)

Sind Rezepte Algorithmen?

Zutaten: 1 Packung Löffelbiskuit, 500g Mascarpone, 4 Eier, 2 Esslöffel Zucker, 5 Esslöffel Amaretto, etwas Kakaopulver, 1 Tasse Kaffee

REZEPT_TIRAMISU

1. Den Kaffee in einen Suppenteller gießen.
2. Die Biskuits kurz im Kaffee tränken.
3. Den Boden einer Auflaufform mit einer Lage Biskuits belegen.
4. Eigelb und Eiweiß trennen.
5. Das Eigelb mit dem Zucker und Amaretto zu schaumiger Masse schlagen.
6. Mascarpone zugeben und gut mischen, bis es cremig wird.
7. Das Eiweiß in einem anderen Behälter steif schlagen.
8. Das geschlagene Eiweiß zur Crème hinzugeben.
9. Die Biskuits in der Form gleichmäßig mit der Hälfte der Crème überziehen.
10. Nun eine neue Lage kaffeegetränkter Biskuits auflegen.
11. Den Rest der Crème gleichmäßig auftragen.
12. Die Form mindestens drei Stunden (besser über Nacht) in den Kühlschrank stellen.
13. Die Crème vor dem Servieren mit Kakao bestreuen.

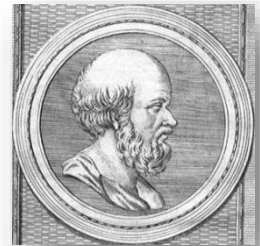


END_REZEPT_TIRAMISU

Ergebnis: Ein leckeres Tiramisu

Sieb des Eratosthenes

- Bestimmung aller Primzahlen bis zu einem gegebenen Wert k
- **Eingabe:** Wert $k \in \mathbb{N}$ (obere Schranke)
- **Ausgabe:** Alle Primzahlen bis zur Schranke k
- **Arbeitsweise** des Algorithmus:



Entfernen der Vielfachen von 2

	②	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48		

Entfernen der Vielfachen von 3

	②	③	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48		

Entfernen der Vielfachen von 5

	②	③	4	⑤	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48		

Die Primzahlen bis 48

	②	③	4	⑤	6	⑦	8	9	10
11	12	⑬	14	15	16	⑰	18	19	20
21	22	⑲	24	25	26	27	28	⑳	30
31	32	33	34	35	36	⑳	38	39	40
41	42	④③	44	45	46	④⑦	48		

Quelle: Logofatu - Grundlegende Algorithmen mit Java

Eigenschaften von Algorithmen

- Die Abfolge der einzelnen Verarbeitungsschritte muss eindeutig aus einem Algorithmus hervorgehen
- Ein Algorithmus ist unabhängig von einer Notation (z.B. natürliche Sprache, Pseudocode, C, C++, Java, C#, ...)
- Es gibt viele Beschreibungen desselben Algorithmus
- Hauptziel beim Entwurf von Algorithmen
 - Korrekte Problemlösung (**totale Korrektheit**):
 - **Terminiertheit**: Der Algorithmus endet für jede spezifizierte Eingabe
 - **Partielle Korrektheit**: Der Algorithmus liefert für jede spezifizierte Eingabe das geforderte Ergebnis
- Nebenziel beim Entwurf von Algorithmen
 - Effiziente Problemlösung hinsichtlich **Zeit** und **Platz**
- Algorithmen sollten effektiv sein

Beispiele für Algorithmen

- Addieren, Subtrahieren, Multiplizieren, Dividieren, etc.
- Kochrezepte, Bastel-/Gebrauchsanleitungen, Spielregeln, ...
- Sortieren, Suchen, Effizienter Umgang mit Datenstrukturen, ...
- Vorsicht: Forderung der Eindeutigkeit fordert Präzision!
- Algorithmus zur Bestimmung des **größten gemeinsamen Teilers (ggT)** zweier natürlicher Zahlen (Euklidischer Algorithmus, ~ 300 v. Chr.):
 - **Eingabe:** $a, b \in \mathbb{N}$ (zwei natürliche Zahlen a und b)
 - **Ausgabe:** $a = \text{ggT}(a, b)$ (ggT von a und b)
 - **Algorithmus:** Wiederhole
 - $r = \text{Rest der ganzzahligen Division von } a/b$
 - $a = b$
 - $b = r$
 Bis $r = 0$ ist
Gib a aus

Nachweis der totalen Korrektheit?

Vorbedingung: $a, b \in \mathbb{N}$; sei G größter gemeinsame Teiler von a und b , $E = x + y$

Nachbedingung: $x = G$

$\{ G \text{ ist ggT}(a, b) \wedge a \in \mathbb{N}^+ \wedge b \in \mathbb{N}^+ \}$

$x := a; y := b;$

$\{ \text{INV: } G \text{ ist ggT}(x, y) \wedge x \in \mathbb{N}^+ \wedge y \in \mathbb{N}^+ \wedge x + y > 0 \}$

solange $x \neq y$ **wiederhole**

$\{ \text{INV} \wedge x \neq y \wedge x + y = k \}$

falls $x > y$:

$\{ G \text{ ist ggT}(x, y) \wedge x > y \wedge x \in \mathbb{N}^+ \wedge y \in \mathbb{N}^+ \wedge x + y = k \} \Rightarrow$

$\{ G \text{ ist ggT}(x - y, y) \wedge x - y \in \mathbb{N}^+ \wedge y \in \mathbb{N}^+ \wedge x - y + y = k - y < k \}$

$x := x - y$

$\{ \text{INV} \wedge x + y < k \}$

sonst

$\{ G \text{ ist ggT}(x, y) \wedge y > x \wedge x \in \mathbb{N}^+ \wedge y \in \mathbb{N}^+ \wedge x + y = k \} \Rightarrow$

$\{ G \text{ ist ggT}(x, y - x) \wedge x \in \mathbb{N}^+ \wedge y - x \in \mathbb{N}^+ \wedge x + y - x = k - x < k \}$

$y := y - x$

$\{ \text{INV} \wedge x + y < k \}$

$\{ \text{INV} \wedge x + y < k \}$

$\{ \text{INV} \wedge x = y \} \Rightarrow \{ x = G \}$

Zentraler Begriff: Datenstruktur

- Algorithmen verarbeiten Eingaben zu Ausgaben und benutzen dabei ggf. unstrukturierte (skalare) und strukturierte Daten
- Datentyp
 - Menge von **Objekten** mit darauf definierten **Operationen**
 - Zwei Varianten:
 - **(konkreter) Datentyp** (abhängig von Rechner und Implementierung)
 - **abstrakter Datentyp** (abstrahiert von spezieller Implementierung)
- Datentyp wird durch folgendes Tupel festgelegt:
 - Objektmenge (Werte)
 - Operationen, definiert durch
 - Signatur (Name mit Definitions- und Wertebereich, **Syntax**)
 - Regeln/Axiome (Wirkung der Operationen, **Semantik**)
- Datenstruktur (= strukturierte Daten + Operationen)
 - **Menge von Datentypen, zwischen denen Beziehungen bestehen**

Beispiele für Datentypen in C++ (32-Bit)

Datentyp	Bits	Wertebereich	Typische Operationen
char, signed char	8	-128 ... 127	+, -, *, /, <, >, %
unsigned char	8	0 ... 255	+, -, *, /, <, >, %
short, signed short	16	-32768 ... 32767	+, -, *, /, <, >, %
unsigned short	16	0 ... 65535	+, -, *, /, <, >, %
int, signed int	32	-2.147.483.648 ... 2.147.483.647	+, -, *, /, <, >, %
unsigned, unsigned int	32	0 ... 4.294.967.295	+, -, *, /, <, >, %
long, signed long	32	-2.147.483.648 ... 2.147.483.647	+, -, *, /, <, >, %
unsigned long	32	0 ... 4.294.967.295	+, -, *, /, <, >, %
float	32	$1,2 \cdot 10^{-38} \dots 3,4 \cdot 10^{38}$	+, -, *, /, <, >
double	64	$2,2 \cdot 10^{-308} \dots 1,8 \cdot 10^{308}$	+, -, *, /, <, >
long double	96	$3,4 \cdot 10^{-4932} \dots 1,1 \cdot 10^{4932}$	+, -, *, /, <, >

- Lernziele

- Grundlegende Algorithmen und Datenstrukturen für Standard-Probleme kennenlernen und implementieren können
- Effizienz von Algorithmen und Datenstrukturen bewerten und vergleichen können
- Effiziente Algorithmen und Datenstrukturen (anhand von kennengelernten Entwurfsprinzipien) entwerfen können

- Inhalt

- Grundlagen
 - Begriffe, Komplexität, Landau-Notation, Rekursionsgleichungen
- Algorithmen
 - Einfügen, Entfernen, Suchen und Sortieren
 - Entwurfsmethoden, Ausgewählte Algorithmen
- Datenstrukturen
 - Listen, Stapel und Schlangen
 - Bäume und Graphen



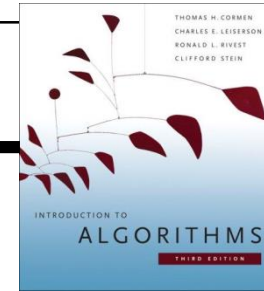
Wechselwirkung

- Schwerpunkt liegt auf **algorithmischen Aspekten**
- Beispielsprachen:
 - Pseudocode, C, C++, Java, C#
(abstrakte, möglichst klare und präzise Beschreibung der Ideen)
- Softwaretechnische Aspekte werden vernachlässigt, z.B.
 - Vollständigkeit
 - Fehlerbehandlung
 - Objekt-orientierter Entwurf, Modularität
 - C, C++, Java, C# - spezifische Konstrukte
 - ...

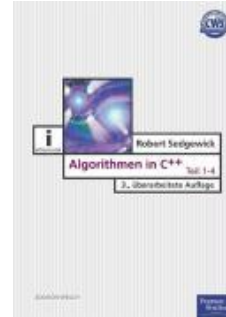
Programmieren im Kleinen

Literatur zur Vorlesung

Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.:
Introduction to Algorithms, 3rd Ed., MIT Press, 2009



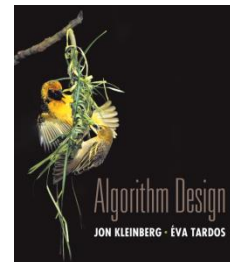
Sedgewick, R.:
Algorithmen in C++, Pearson Studium, 3. Auflage, 2002



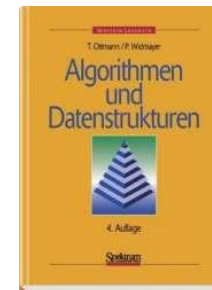
Schöning, U.:
Algorithmik, Spektrum Akademischer Verlag, 2001



Kleinberg, J., Tardos, É.:
Algorithm Design, Addison Wesley, 2005

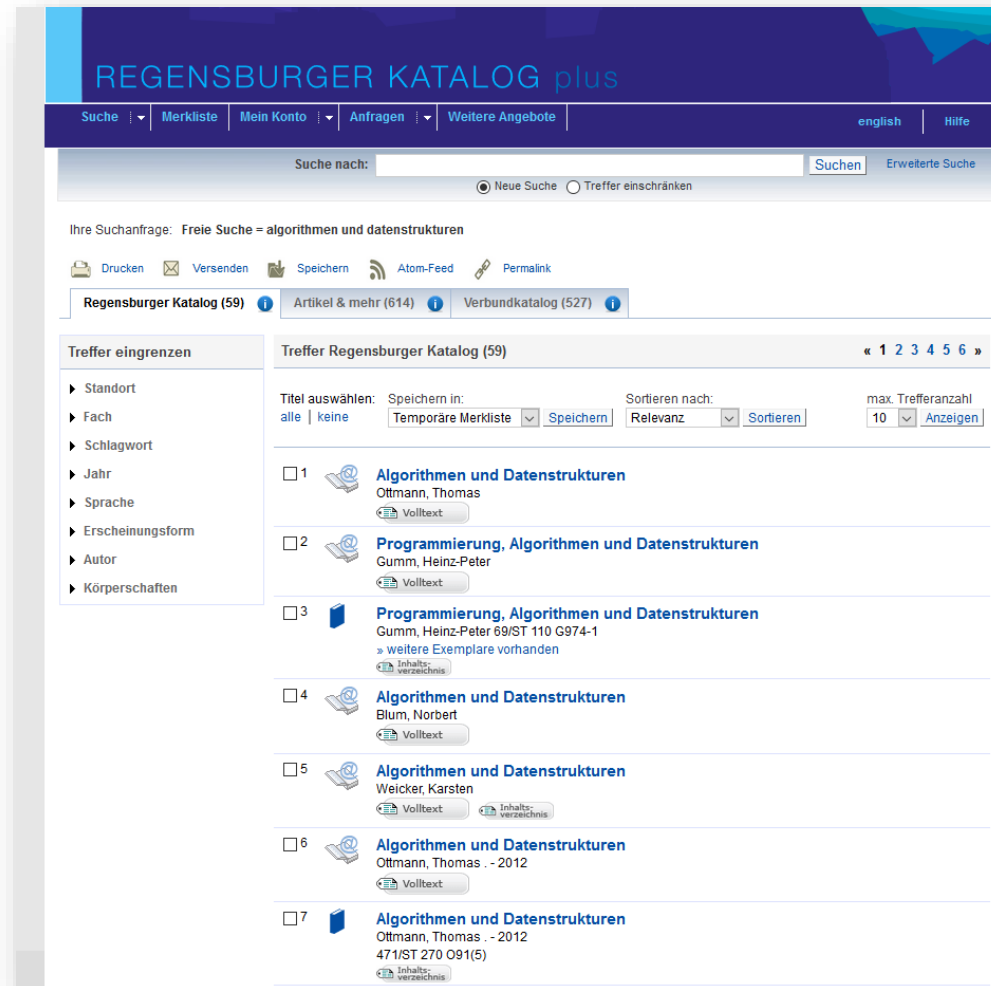


Ottmann, T., Widmayer, P.:
Algorithmen und Datenstrukturen, Spektrum Akademischer Verlag, 4. Auflage, 2002



Pomberger, G., Dobler, H.:
Algorithmen und Datenstrukturen, Pearson Studium 2008





<https://www.regensburger-katalog.de>