



---

12. April 2023

---

*Automaten und Formale Sprachen* ist der Name unserer Fachgruppe innerhalb der Gesellschaft für Informatik; daraus leitet sich auch das Akronym unserer in lockerer Folge veröffentlichten Rundschreiben ab. Dies ist der zweite Rundbrief im Jahr 2023.

Dieser Rundbrief enthält eine Ankündigung zum diesjährigen Theorietag (mit mehr als dem bereits verkündeten Datum und Ort) und Ankündigungen oder Neuigkeiten zu weiteren Konferenzen (DLT, WORDS, DCFS, NCMA und CIAA). Außerdem listen wir wieder bevorstehende Tagungen mit nahem Einsendeschluss auf.

Dieses Mal haben wir auch einen Bericht aus einer Arbeitsgruppe, nämlich von Andreas Mühling (Christian-Albrechts-Universität zu Kiel und IPN – Leibniz Institut für die Pädagogik der Naturwissenschaften und Mathematik, Kiel) zum Thema Didaktik der theoretischen Informatik.

## Tagungsankündigungen

In der Reihenfolge ihres Auftretens möchten wir hier auf ein paar Konferenzen besonders eingehen (und Neuigkeiten zu bereits im vorherigen Rundbrief angekündigten Tagungen bekannt geben): DLT und WORDS (12.–16. Juni in Umeå), DCFS (4.–6. Juli in Potsdam), NCMA (18.–19. September in Famagusta), CIAA (19.–22. September in Famagusta) und Theorietag (4.–6. Oktober in Kaiserslautern).

### DLT und WORDS

Von Frank Drewes

12.–16. Juni 2023, Umeå (Schweden)

Vom 12.–16. Juni 2023 werden in Umeå (Schweden) die internationalen Konferenzen Developments in Language Theory und WORDS stattfinden.

Die Liste angenommener Papiere ist jetzt auf der Homepage  
<https://dltwords2023.cs.umu.se/>  
veröffentlicht und die „early registration“ wird bis zum 30. April möglich sein, siehe  
<https://dltwords2023.cs.umu.se/index.php/registration/>.

Wir sind besonders froh darüber, ankündigen zu können, dass Moshe Y. Vardi, Rice University, nach Umeå kommen wird, um den „Salomaa Prize in Automata Theory,

Formal Languages and Related Topics“ aus der Hand von Wolfgang Thomas entgegen zu nehmen. Für weitere Details verweisen wir auf die Information auf der DLT&WORDS-Homepage.

## DCFS

Von Henning Bordihn

4.–6. Juli 2023, Potsdam

Die internationale Konferenz zur Beschreibungskomplexität formaler Systeme (Descriptive Complexity of Formal Systems, DCFS) wird vom 4.–6. Juli 2023 in Potsdam stattfinden. Es ist die 25. Ausgabe dieser Reihe. Sie wird vom Institut für Informatik und Computational Science der Universität Potsdam und der IFIP Working Group 1.02 „Descriptive Complexity“ organisiert.

Auf der Konferenz-Homepage

<https://www.cs.uni-potsdam.de/dcfs2023/>

sind jetzt neu veröffentlicht eine Liste der angenommenen Arbeiten und eine Liste von eingeladenen Sprechern: Pascal Caron (Rouen), Friedrich Otto (Kassel), Rogério Reis (Porto) sowie Jürgen Dassow (Magdeburg), wobei letzterer eine Festrede anlässlich der 25. Ausgabe der DCFS halten wird.

Der Zeitraum für „early registration“ läuft bis zum 14. Mai. Weitere Angaben findet man auf genannter Homepage.

## NCMA

Von Benedek Nagy

18.–19. September 2023, Famagusta (Nordzypern)

Das Ziel der Workshops zu „Non-Classical Models of Automata and Applications“ ist, ein Forum für Forscher zu bieten, die an verschiedenen Aspekten klassischer und nichtklassischer Automatenmodelle und Grammatiken arbeiten, um neue Ideen auszutauschen und zu entwickeln.

Um den Workshop-Charakter der NCMA 2023 zu betonen, laden wir dazu ein, zusätzlich zu vollwertigen Beiträgen auch Kurzbeiträge zu jüngsten Resultaten oder noch laufender Arbeit einzureichen. Im Gegensatz zu den akzeptierten regulären Beiträgen, die im offiziellen Tagungsband veröffentlicht werden, erscheinen die akzeptierten Kurzbeiträge in einem separaten technischen Bericht. Im Anschluss ist ein Sonderheft bei *Acta Informatica* mit ausgewählten und erweiterten NCMA-Beiträgen geplant.

Wichtige Daten:

Einsendeschluss für Vollversionen: 31. Mai 2023

Einsendeschluss für Kurzbeiträge: 3. Juli 2023

Benachrichtigung: 10. bzw. 12. Juli 2023

Endfassung: 24. Juli 2023

Konferenz: 18.–19. September 2023

Für Hauptvorträge wurden Friedrich Otto (Kassel) und György Vaszil (Debrecen) gewonnen. Weiteres gibt es auf der Seite

<https://ncma.emu.edu.tr/>.

## CIAA

Von Benedek Nagy

19.–22. September 2023, Famagusta (Nordzypern)

Die 27. internationale Konferenz zu Implementation und Anwendung von Automaten (Conference on Implementation and Application of Automata, CIAA) findet im Anschluss an die NCMA (siehe oben) am gleichen Ort statt. Beide Tagungen werden organisiert von der Eastern Mediterranean University in Famagusta (Nordzypern).

Die CIAA-Konferenzreihe deckt alle Aspekte von Implementation, Anwendung und Theorie von Automaten und verwandten Strukturen ab. Ihr Ziel ist, Beiträge sowohl aus der klassischen Automatentheorie als auch zu Anwendungen anzuziehen. Der Tagungsband wird vom Springer-Verlag in der Reihe *Lecture Notes in Computer Science* veröffentlicht. Für ausgewählte und erweiterte Konferenzbeiträge ist ein Sonderheft des *International Journal of Foundations of Computer Science* (IJFCS) geplant.

Wichtige Daten:

Einsendeschluss: 14. April 2023

Benachrichtigung: 29. Mai 2023

Endfassung: 9. Juni 2023

Konferenz: 19.–22. September 2023

Auf der Seite

<https://ciaa.emu.edu.tr/>

findet man neben weiteren Informationen auch Kurzfassungen der Hauptvorträge, die von Viliam Geffert (Košice), Friedrich Otto (Kassel) und Cem Say (Istanbul) gehalten werden.

## Theorietag

Von Anthony W. Lin und Georg Zetsche

4.–6. Oktober 2023, Kaiserslautern

Der Theorietag „Automaten und Formale Sprachen“, welcher auch das Jahrestreffen unserer Fachgruppe ist, wird vom 4. bis zum 6. Oktober 2023 in Kaiserslautern stattfinden. Die Liste der eingeladenen Vortragenden steht fest und die Website ist fertig:

<https://theorietag2023.mpi-sws.org/>

Besagte Hauptvorträge werden gehalten von Christoph Haase (Oxford), Sandra Kiefer (Oxford), Daniel Neider (Dortmund), Joël Ouaknine (Saarbrücken) und Anne-Kathrin Schmuck (Kaiserslautern).

Der Theorietag wird organisiert von Anthony W. Lin (RPTU Kaiserslautern) und Georg Zetsche (Max-Planck-Institut für Software-Systeme, MPI-SWS). Er wird im Gebäude des MPI-SWS stattfinden. Unterkünfte sind selbst zu buchen. Eine Auswahl an Möglichkeiten gibt es auf der genannten Homepage, ebenso eine Anreisebeschreibung.

## Tagungen mit nahendem Einsendeschluss

Die Tagungen der folgenden Liste sind nach Datum des Einsendeschlusses sortiert. Die Veranstaltungsreihenfolge ist oft eine andere.

- CIAA 2023 — Einsendeschluss: 14. April 2023  
27. International Conference on Implementation and Application of Automata  
<https://ciaa.emu.edu.tr/>  
19.–22. September 2023, Famagusta, Nordzypern
- AUTOMATA 2023 — Einsendeschluss: 21. April 2023  
29. International Workshop on Cellular Automata and Discrete Complex Systems  
<http://automata2023.units.it/>  
30. August–1. September 2023, Trieste, Italien
- MFCS 2023 — Einsendeschluss: 24. April 2023 (Kurzfassungen), 28. April 2023 (Langfassungen)  
48. International Symposium on Mathematical Foundations of Computer Science  
<https://mfcs2023.labri.fr/>  
28. August–1. September 2023, Bordeaux, Frankreich
- SPIRE 2023 — Einsendeschluss: 19. Mai 2023  
30. International Symposium on String Processing and Information Retrieval  
<http://spire2023.isti.cnr.it/>  
26.–28. September 2023, Pisa, Italien
- AFL 2023 — Einsendeschluss: 28. Mai 2023  
16. International Conference on Automata and Formal Languages  
<https://afl2023.uni-eszterhazy.hu/>  
5.–7. September 2023, Eger, Ungarn
- FCT 2023 — Einsendeschluss: 26. Mai 2023 (Kurzfassungen), 1. Juni 2023 (Langfassungen)  
24. International Symposium on Fundamentals of Computation Theory  
<https://www.uni-trier.de/index.php?id=71937&L=2>  
18.–21. September 2023, Trier, Deutschland  
Lila Kari hält einen Hauptvortrag.
- NCMA 2023 — Einsendeschluss: 31. Mai 2023  
13. International Workshop on Non-Classical Models of Automata and Applications  
<https://ncma.emu.edu.tr/>  
18.–19. September 2023, Famagusta, Nordzypern
- ICTAC 2023 — Einsendeschluss: 16. Juni 2023  
20. International Colloquium on Theoretical Aspects of Computing  
<https://ictac2023.compsust.utec.edu.pe/>  
4.–8. Dezember 2023, Lima, Peru

- WAFNL 2023 — Einsendeschluss 20. Juni 2023 (Kurzfassungen), 27. Juni 2023 (Langfassungen)  
Workshop on Automata, Formal Languages, Natural Language Processing, and Computational Linguistics  
<https://itat.ics.upjs.sk/index.php?id=ws/WAFNL>  
23.–25. September 2023, Tatranské Matliare, Slowakei
- GandALF 2023 — Einsendeschluss 23. Juni 2023 (Kurzfassungen), 30. Juni 2023 (Langfassungen)  
14. International Symposium on Games, Automata, Logics, and Formal Verification  
<https://gandalf23.uniud.it/>  
18.–20. September, Udine, Italien
- Theorietag 2023 — Einsendeschluss: 15. August 2023  
33. Theorietag „Automaten und Formale Sprachen“  
<https://theorietag2023.mpi-sws.org>  
4.–6. Oktober 2023, Kaiserslautern, Deutschland
- SOFSEM 2024 — Einsendeschluss: 28. August 2023 (Kurzfassungen), 1. September 2023 (Langfassungen)  
49. International Conference on Current Trends in Theory and Practice of Computer Science  
<https://www.uni-trier.de/index.php?id=90670&L=2>  
19.–23. Februar 2024, Cochem, Deutschland  
Markus Schmid und Sandra Zilles halten zwei der Hauptvorträge.

## Berichte aus Arbeitsgruppen

Dieses Mal von Andreas Mühling (Christian-Albrechts-Universität zu Kiel und IPN – Leibniz Institut für die Pädagogik der Naturwissenschaften und Mathematik, Kiel):

Wie vermittelt man theoretische Informatik und warum? Jedes Fach hat die Themen, die von außen betrachtet als „schwer“ gelten. Denen der Ruf voraneilt, dass man „da eben durch“ muss, die vielleicht überhaupt nur wegen des „Aussiebens“ im Stundenplan eines Studiengangs stehen. Einige – meistens eher wenige – finden sich dann aber doch, die Begeisterung für diese Themen entwickeln, die sich darauf spezialisieren und dann manchmal auch selbst wieder vor der Frage stehen, wie man diese Themen denn nun so vermittelt, dass die eigene Begeisterung überspringt und dass man das eigene Thema aus der Ecke der „da muss man halt durch“-Themen herausholt. Aus didaktischer Sicht sind diese Situationen die Interessanteren, denn „leichte“ Themen oder Themen bei denen die Lernenden sowieso voller Elan bei der Sache sind, lassen sich auch leicht vermitteln. Situationen wiederum, bei denen Lehrende und Lernende gleichermaßen keine Begeisterung aufbringen können, sind andererseits auch von vornherein zum Scheitern verurteilt. Was kann also die Fachdidaktik der Informatik – das Bindeglied zwischen der allgemeinen Lehr-Lernforschung und der Fachwissenschaft – zum Thema der theoretischen Informatik in der (universitären) Lehre beitragen? Darum soll es im Folgenden gehen. Im Gegensatz zur Mathematik, in der es bereits eine Statistikdidaktik oder eine Didaktik der euklidischen Geometrie gibt,

haben wir in der Informatik bisher keine auf verschiedene Bereiche ausdifferenzierte Didaktik. Dennoch gibt es fachdidaktische Forschungs- und Entwicklungsarbeiten in den verschiedenen Teilgebieten der Informatik, meist auch mit einem Fokus auf die Hochschule, da die Schulinformatik bisher nur eine vergleichsweise geringe Bedeutung hat. Fast man die fachdidaktischen Erkenntnisse zur „theoretischen Informatik“ (im weitesten Sinne) zusammen, so muss man zunächst feststellen, dass es hier weit weniger Arbeiten – und noch weniger empirisch belastbare Ergebnisse – gibt als etwa im Bereich der Programmierung. Belastbare Ergebnisse entstehen über die Zeit durch die Zusammenfassung von Einzelstudien und deren Replikationen in Metastudien oder ggf. auch der Zusammenfassung von Metastudien in Metareviews. Fachgebietsspezifische, didaktische Modelle oder „Best-Practices“ zur Didaktik der theoretischen Informatik sind also leider bisher rar gesät, so dass neben vielen einzelnen Forschungsarbeiten nur ein Blick in allgemeine Mechanismen des Lehrens und Lernens bleibt. Beides wird im Folgenden miteinander kombiniert an einigen Beispielen dargestellt. Es ist zunächst festzuhalten, dass die theoretische Informatik nach typischen maßgeblichen Merkmalen tatsächlich ein „schweres“ Thema ist. Ein solches Merkmal ist die hohe themenbezogene kognitive Last, die durch die dichte Vernetzung der am Thema beteiligten Konzepte entsteht [11]. Dadurch bleibt vergleichsweise weniger kognitive Kapazität für das Lernen an sich (die lernbezogene Last) übrig und der Lernprozess gestaltet sich schwieriger. Diese hohe Last zeigt sich auch zum Beispiel darin, dass alle Studierenden, unabhängig von ihrer Endnote bei bestimmten Themen einer Theorieveranstaltung – speziell dem Verfassen von formalen Beweisen – Probleme haben [4]. Aus diesem Grund ist die Art, wie man das Lernmaterial darstellt, in der Theorie besonders wichtig, da sich hier die dritte – und die am einfachsten zu beeinflussende – Stellschraube für kognitive Last befindet – die materialbezogene Last. Hier gibt es zum Beispiel einen negativen Effekt bei der redundanten Darstellung von Informationen („redundancy effect“) oder einen negativen Effekt bei der Aufteilung von Informationen etwa in Text- und Diagrammform („split-attention effect“). Stellt man also zum Beispiel einen endlichen Automaten in Form eines Übergangsdiagramms und in Tupel-Schreibweise dar, hat man bereits beide Effekte vorliegen. Typische Maßnahmen, um einer hohen themenbezogenen Belastung entgegenzuwirken, sind, den Blick vom Kleinen auf’s Große zu richten – anstelle das große Ganze bereits von Anfang an mit im Blick zu haben („isolated interacting elements effect“). Auch Visualisierungen, speziell in einer Form, die man auch „vor dem geistigen Auge“ nachvollziehen kann, helfen dabei („imagination effect“). Die Darstellung eines Automaten als Übergangstabelle ist dabei z. B. wenig hilfreich. Das Zusammenarbeiten an Aufgaben im Sinne des „Pair-Programming“ kann dabei auch in der theoretischen Informatik helfen [1]. Ein zweites Merkmal für „schwere“ Themen ist, dass diese – zum Beispiel durch ihren abstrakten Charakter – wenig Anknüpfungspunkte an Vorwissen bieten [7] – ein ganz entscheidender Faktor für erfolgreiches Lernen. Auch fällt es bei solchen Themen oft schwer, die Relevanz richtig einzuordnen, eine wichtige Stellschraube für Lernmotivation. Dadurch kann ein Teufelskreis aus anfänglich fehlenden Erfolgserlebnissen und darauf aufbauend geringerer Motivation, die wiederum zu geringeren Erfolgen führt, entstehen, bis letztlich der resignierende Eindruck „das Thema kann ich einfach nicht“ bleibt. Hier sind die Lösungen letztlich sehr einfach: Erfolgserlebnisse ermöglichen – zum Beispiel durch den Einsatz von digitalen Tools [6, 8, 2], um dadurch die Motivation zu erhöhen und die Themen möglichst eingebettet, zum Beispiel in Anwendungskontexte [5], präsentieren, so dass die Relevanz und ggf. auch relevan-

tes Vorwissen sofort erkennbar werden. Es macht motivational einen Unterschied, ob man mit einer Grammatik alle Wörter über dem Alphabet  $\{a, b\}$  erkennen möchte, die die Form  $a^n b^n$  haben, oder ob man über das Problem nachdenkt, wie man eine korrekte Klammerung in einem Taschenrechner erkennen kann. Ein neueres Beispiel für die Einbettung theoretischer Themen in den Kontext des Programmierens findet sich bei [3]. Erwachsene Lernende sind in der Lage, aus dem Wissen über die grundsätzliche Sinnhaftigkeit eines Themas für sich Motivation zu generieren, die zwar weniger stark wirkt als die inhärente Freude am Thema, aber dennoch stärker als die drohende Klausur (vgl. [9]). Es ist hier bereits offensichtlich, dass sich solche allgemeinen Erkenntnisse immer in einem Spannungsfeld befinden. Das Einbetten in einen Kontext erhöht die Motivation erzeugt aber gleichzeitig auch mehr kognitive Last. Eine grundsätzliche Abwägung des Für und Wider ist dabei nicht möglich – es hängt immer von den Details des Einzelfalls ab. Das Ziel sollte ein „best of both worlds“-Ansatz sein, im Beispiel also die Verwendung eines nicht zu aufwändigen Kontextes, der ggf. auch nur als Einstieg und Ausstieg in ein Thema dient („De- und Rekontextualisierung“). Auf diesem Weg lässt sich auch vermeiden, dass man durch eine „neue Lehre“ plötzlich die Studierenden verliert, die auch bisher schon in der theoretischen Informatik gut zurechtgekommen sind. Stellt man ganz abseits von der Frage nach schweren Themen, die Frage nach guter (universitärer) Lehre, so haben [10] vor einigen Jahren eine umfangreiche Meta-Analyse der Effekte von verschiedenen Faktoren und Maßnahmen vorgelegt. Betrachtet man die Top 5 für den Bereich „Instruction“, so findet man an Platz 1 die Vorbereitung und Organisiertheit der Lehrperson, gefolgt von 2) einer klaren Darstellungsweise, 3) der Möglichkeit für Lernende, ihre eigene Leistung zu reflektieren („self-assessment“), 4) die Förderung von Interesse an Themen und 5) die Förderung des Fragestellens. Alle diese Effekte sind dabei noch weit überdurchschnittlich in ihrer Stärke, das heißt, es lohnt sich, für eine Verbesserung der Lehrqualität hier zu investieren, da dadurch grundsätzlich alle profitieren.

## Literatur

- [1] Oana Andrei (2022) *A Practice Enquiry Design to Investigate How Pair Programming Can Help with Constructing Automata*, Association for Computing Machinery, UKICER '22, in Proceedings of the 2022 Conference on United Kingdom & Ireland Computing Education Research
- [2] Carlos I. Chesñevar, María L. Cobo, William Yurcik (2003) *Using Theoretical Computer Simulators for Formal Languages and Automata Theory*, 33–37, 35, 2, ACM SIGCSE Bulletin
- [3] Matthew Farrugia-Roberts, Bryn Jeffries, Harald Sndergaard (2022) *Programming to Learn: Logic and Computation from a Programming Perspective*, 311–317, Association for Computing Machinery, ITiCSE '22, in Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1
- [4] Christiane Frede, Maria Knobelsdorf (2018) *Exploring How Students Perform in a Theory of Computation Course Using Final Exam and Homework Assignments Data*, 241–249, Association for Computing Machinery, ICER '18, in Proceedings of the 2018 ACM Conference on International Computing Education Research
- [5] Hashim Habiballa, Tibor Kmet' (2004) *Theoretical branches in teaching computer science*, 829–841, 35, 6, International Journal of Mathematical Education in Science and Technology

- [6] Maria Knobelsdorf, Christiane Frede, Sebastian Böhne, Christoph Kreitz (2017) *Theorem Provers as a Learning Tool in Theory of Computation*, 83–92, Association for Computing Machinery, ICER '17, in Proceedings of the 2017 ACM Conference on International Computing Education Research
- [7] Joseph D. Novak and Bob Gowin (2006) *Learning How to Learn (21. Aufl.)*. Cambridge, New York: Cambridge University Press.
- [8] Susan H. Rodger, Bart Bressler, Thomas Finley, Stephen Reading (2006) *Turning Automata Theory into a Hands-on Course*, 379–383, 38, 1, ACM SIGCSE Bulletin
- [9] Richard M. Ryan, Edward L. Deci (2002) *Overview of Self-Determination Theory: An Organismic Dialectical Perspective*, 3–33, University of Rochester Press, editors: Deci, Edward L. and Ryan, Richard M., Handbook of Self-Determination Research
- [10] Michael Schneider, Fanzis Preckel (2017) *Variables Associated with Achievement in Higher Education: A Systematic Review of Meta-analyses*, 565–600, 143, 6, Psychological Bulletin
- [11] John Sweller (2010) *Element Interactivity and Intrinsic, Extraneous, and Germane Cognitive Load*, 123–138, 22, 2, Educational Psychology Review

## Schlusswort

Wir freuen uns auf Ihre Rückmeldungen und Anregungen sowie auf Beiträge zu unserem Rundbrief. Für Nachrichten an die gesamte Fachgruppenleitung oder nur an Sprecher und Stellvertreter dürfen vorzugsweise die generischen e-Mail-Adressen

[fg-afs-leitung@gi.de](mailto:fg-afs-leitung@gi.de) bzw. [fg-afs-sprecher@gi.de](mailto:fg-afs-sprecher@gi.de)

verwendet werden.



Bianca Truthe, Gießen  
(Sprecher)

Florin Manea, Göttingen  
(Stellvertretender Sprecher)

Henning Fernau, Trier

Pamela Fleischmann, Kiel

Markus Schmid, Berlin

Ihre Fachgruppenleitung