

Programmierung 1

Vorlesung 1

Livestream beginnt um 14:15 Uhr

Willkommen

Programmierung 1

Plan für heute

- ▶ **Organisatorisches:** Team, Plattformen, ...
- ▶ **Grundsätzliches:** Ziele, Kursaufbau, ...
- ▶ **Inhaltliches:** Schnellkurs, Teil 1



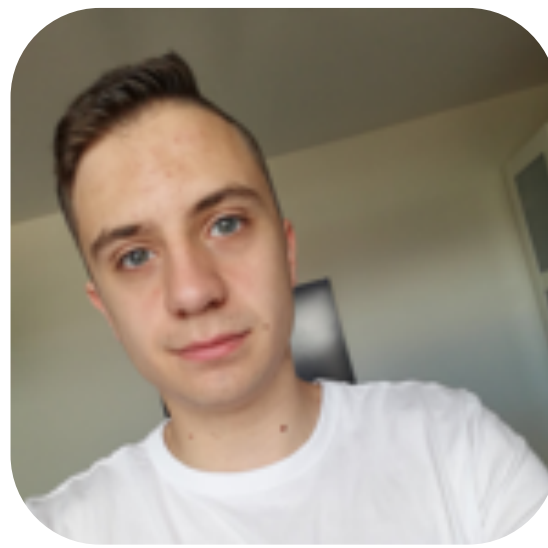
Bernd Finkbeiner



Jana Hofmann



Florian Bies



Paul Eichler



Jacob Graser



Sebastian Holler



Nils Husung



Tim Lukas Janßen



Janine Lohse



Nico Mansion



Gideon Mohr



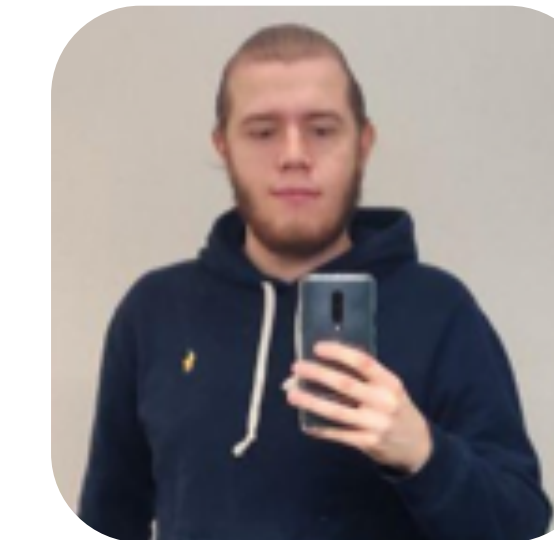
Niklas Mück



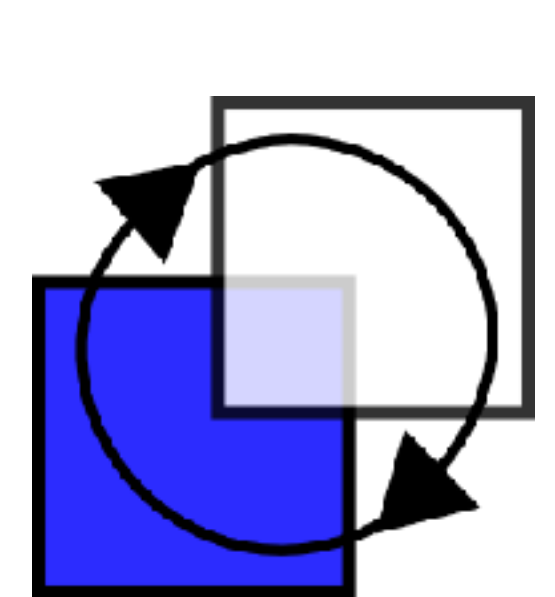
Jonathan Nöther



Benjamin Peters



Ulysse Planta



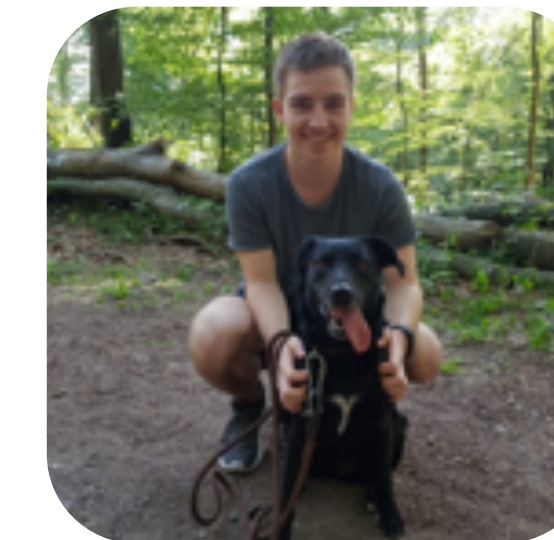
Nina Quach



Fabian Thomas



Yannik Schnitzer



Leon Trampert

Vorlesung

Di 14:15-16:00
Do 10:15-12:00

tv.prog1.saarland

zoom

Tutorium

Slot A: Di 08:15-09:45
Slot B: Di 10:15-11:45
Slot C: Mi 11:15-12:45
Slot D: Mi 16:15-17:45
Slot E: Mi 18:15-19:45

zoom  DISCORD

Office hour

Mo 16:00-18:00, Di 12:00-14:00, Mi 12:00-14:00 Fr 14:00-16:00

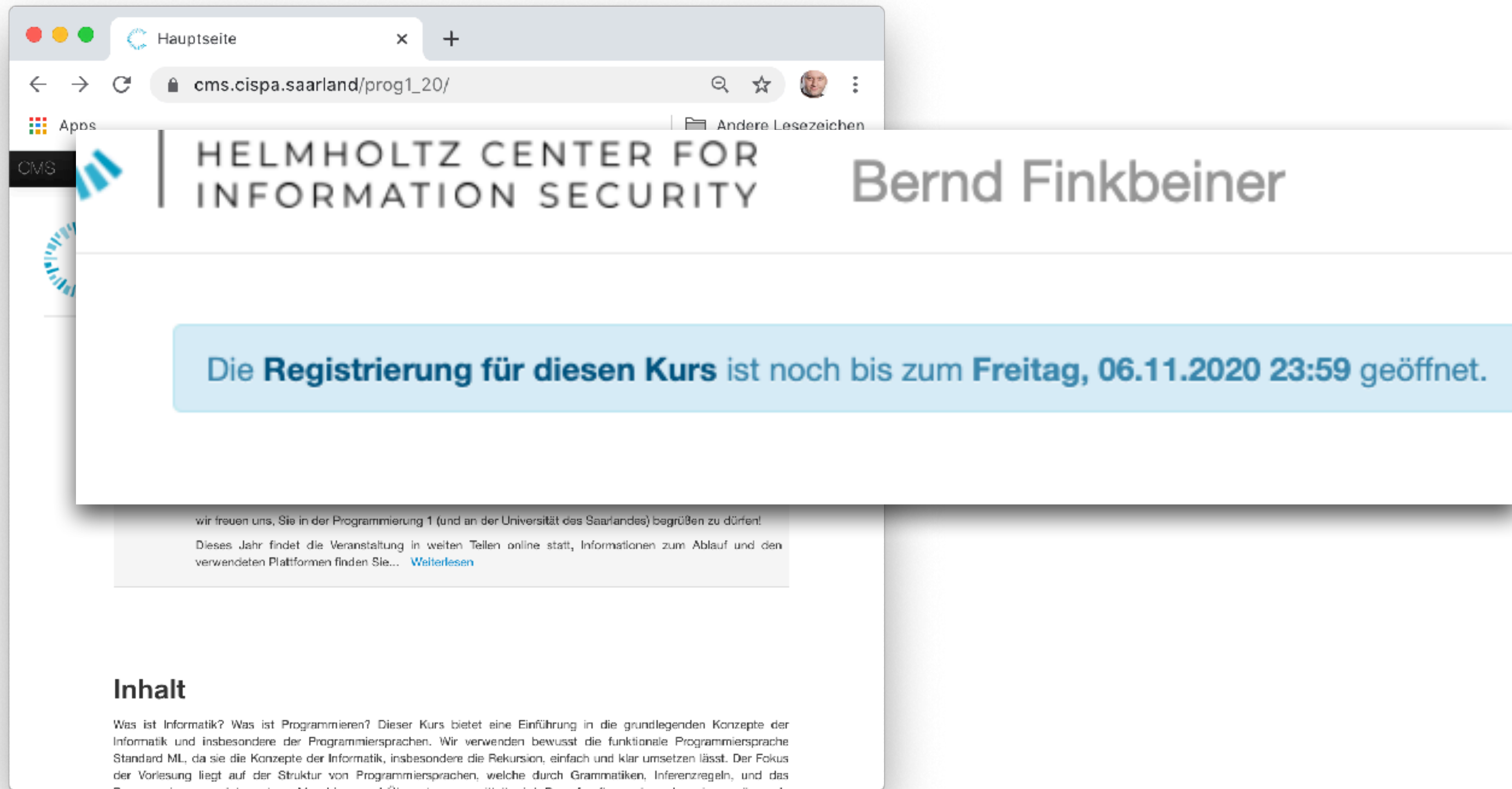
 DISCORD

Minitest

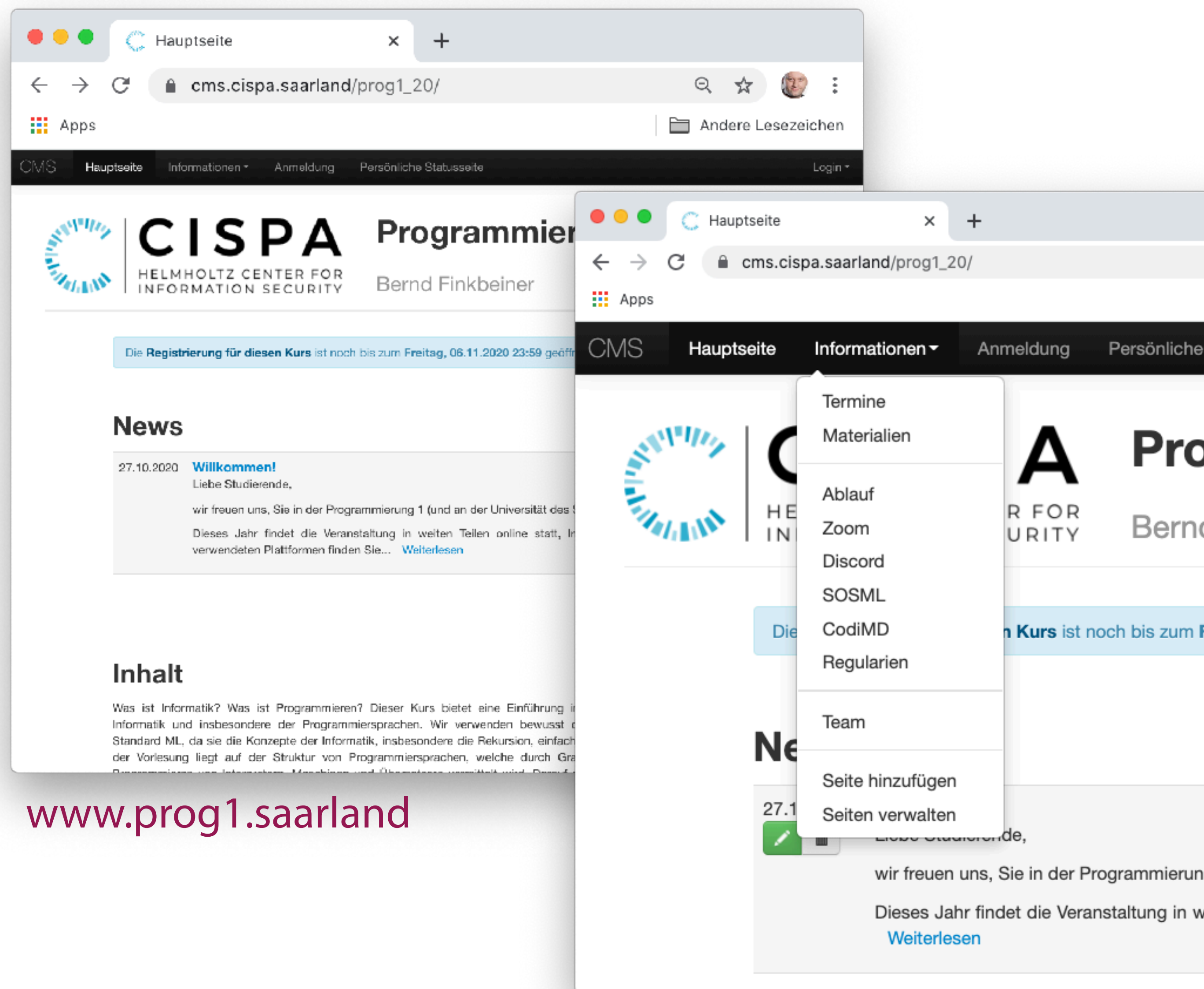
Do 10:00-10:15

minitest.prog1.saarland

ab nächster Woche



www.prog1.saarland



www.prog1.saarland

Hauptseite

cms.cispa.saarland/prog1_20/

Apps

CMSHauptseiteInformationenAnmeldungPersönliche Statusseite



CISPA

HELMHOLTZ CENTER FOR
INFORMATION SECURITY

Programmierung

Bernd Finkbeiner

Die Registrierung für diesen Kurs ist noch bis zum Freitag, 06.11.2020 23:59 geöffnet.

News

27.10.2020

Willkommen!
Liebe Studierende,

wir freuen uns, Sie in der Programmierung 1 (und an der Universität des Saarlandes) b

Dieses Jahr findet die Veranstaltung in weiten Teilen online statt, Informationen
verwendeten Plattformen finden Sie... [Weiterlesen](#)

Inhalt

Was ist Informatik? Was ist Programmieren? Dieser Kurs bietet eine Einführung in die grundl
Informatik und insbesondere der Programmiersprachen. Wir verwenden bewusst die funktiona
Standard ML, da sie die Konzepte der Informatik, insbesondere die Rekursion, einfach und klar um
der Vorlesung liegt auf der Struktur von Programmiersprachen, welche durch Grammatiken, I
Programmierung von Interaktiven Maschinen und Übersetzern vermittelt wird. Darauf aufbauend

Termine

cms.cispa.saarland/prog1_20/termine/calendar/index

Apps

CMSHauptseiteInformationenAnmeldungPersönliche StatusseiteMinitestsSOSMLForumInternKonfigurationLogout



CISPA

HELMHOLTZ CENTER FOR
INFORMATION SECURITY

Programmierung 1

Bernd Finkbeiner

November 2020

HeuteTagWocheMonat<>

Mo	Di	Mi	Do	Fr	Sa	So
26	27	28	29	30	31	1
	14:15 Vorlesung		10:15 Vorlesung	6	7	8
9	10	11	12	13	14	15
08:15 Tutorium Slot A	14:15 Tutorium Slot C	10:15 Vorlesung				
10:15 Tutorium Slot B	16:15 Tutorium Slot D					
14:15 Vorlesung	18:15 Tutorium Slot E					
16	17	18	19	20	21	22
08:15 Tutorium Slot A	14:15 Tutorium Slot C	10:00 Minitest	10:15 Vorlesung			
10:15 Tutorium Slot B	16:15 Tutorium Slot D					
14:15 Vorlesung	18:15 Tutorium Slot E					
23	24	25	26	27	28	29
08:15 Tutorium Slot A	14:15 Tutorium Slot C	10:00 Minitest	10:15 Vorlesung			
10:15 Tutorium Slot B	16:15 Tutorium Slot D					
14:15 Vorlesung	18:15 Tutorium Slot F					
30	1	2	3	4	5	6
08:15 Tutorium Slot A	14:15 Tutorium Slot C	10:00 Minitest	10:15 Vorlesung			
10:15 Tutorium Slot B	16:15 Tutorium Slot D					
14:15 Vorlesung	18:15 Tutorium Slot E					

Terminübersicht

Terminkategorien verwalten

Neuer Termin

Neue Termin-Serie

Sie können diesen Kalender in Ihre Terminverwaltung importieren, indem Sie den Export-Link aus dem "Terminübersicht"-Menü kopieren.

www.prog1.saarland

Hauptseite

cms.cispa.saarland/prog1_20/

Apps

CMS

Hauptseite

Informationen

Anmeldung

Persönliche Status



CISPA
HELMHOLTZ CENTER FOR
INFORMATION SECURITY

Progr

Bernd Fi

Die Registrierung für diesen Kurs ist noch bis zum Freitag, 0

News

27.10.2020 **Willkommen!**
Liebe Studierende,
wir freuen uns, Sie in der Programmierung 1 (und
Dieses Jahr findet die Veranstaltung in weiten
verwendeten Plattformen finden Sie... [Weiterlese](#)

Inhalt

Was ist Informatik? Was ist Programmieren? Dieser Kurs bie
Informatik und insbesondere der Programmiersprachen. Wir
Standard ML, da sie die Konzepte der Informatik, insbesondere
der Vorlesung liegt auf der Struktur von Programmiersprache
Programmieren von Interneten, Maschinen und Übernetzen

Personliche Statusseite

https://cms.cispa.saarland/

Apps

CMS

Hauptseite

Informationen

Anmeldung

Persönliche Statusseite

Minitests

SOSML

Forum

CodiMD

Vor

⇒ Account bearbeiten

Neuigkeiten

Persönliche Daten

Präferenzen

Übungsgruppen

Discord ID

Anonymer Kommentar

Einstellungen

Präferenzen für die Übungsgruppen

Wir benutzen ein automatisches Verfahren um Sie in die Übungsgruppen einzuteilen, das ba
Vorlesungsteilnehmer optimale Einteilung wählt.

Wir versuchen Sie nicht in einen Slot einteilen, der für Sie **Nicht Okay** ist. Wenn möglich, teil
uns das nicht möglich ist, versuchen wir Sie in einen Slot einzuteilen, der für Sie **Okay** ist.

Sie können Ihre Präferenzen noch bis **Freitag, 06.11.2020 23:59** ändern.

Slot 1: Dienstag, 08:15

Preferred

Okay

Not Okay

Slot 2: Dienstag, 10:15

Preferred

Okay

Not Okay

Slot 3: Mittwoch, 14:15

Preferred

Okay

Not Okay

Slot 4: Mittwoch, 16:15

Preferred

Okay

Not Okay

Slot 5: Mittwoch, 18:15

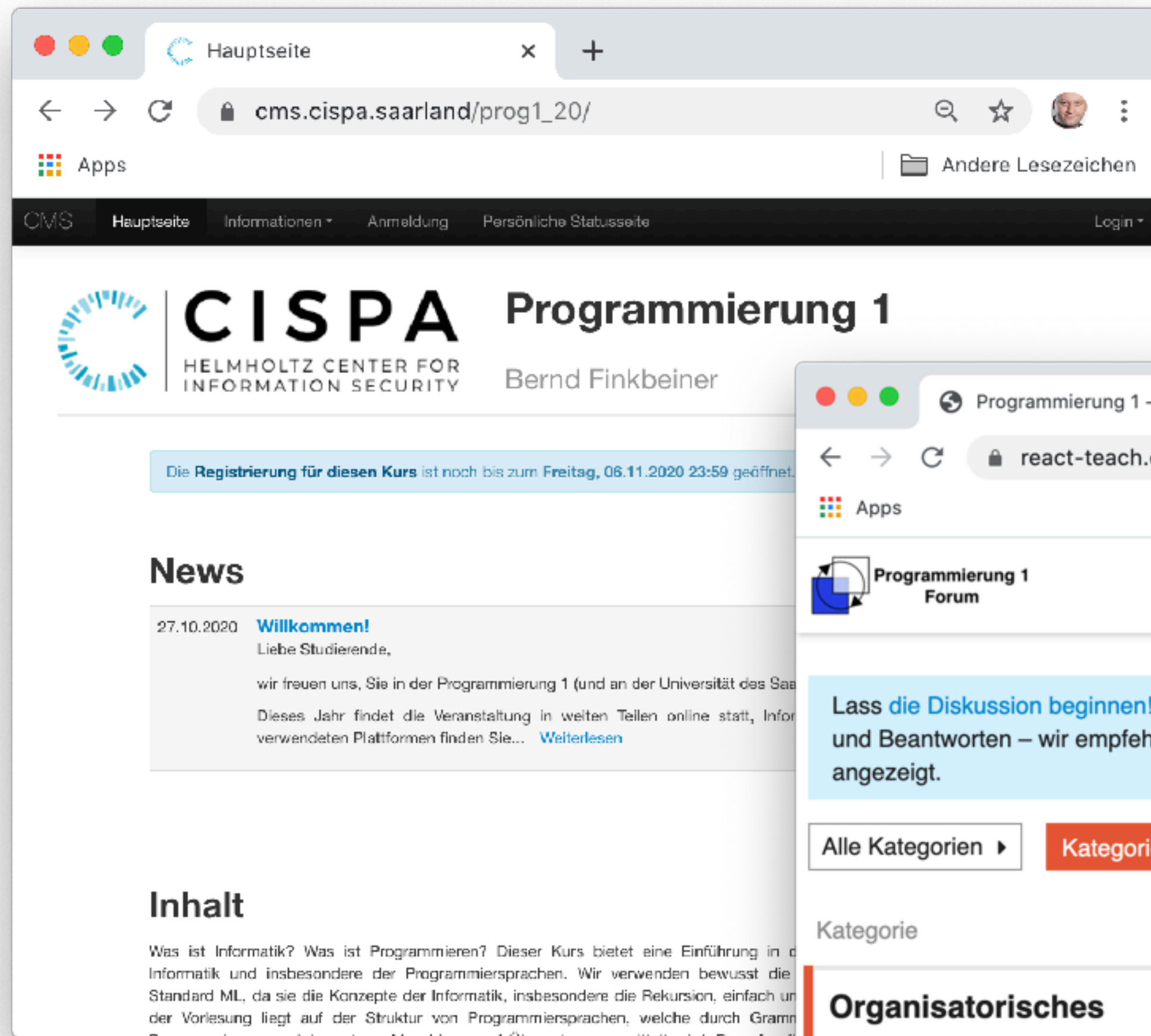
Preferred

Okay

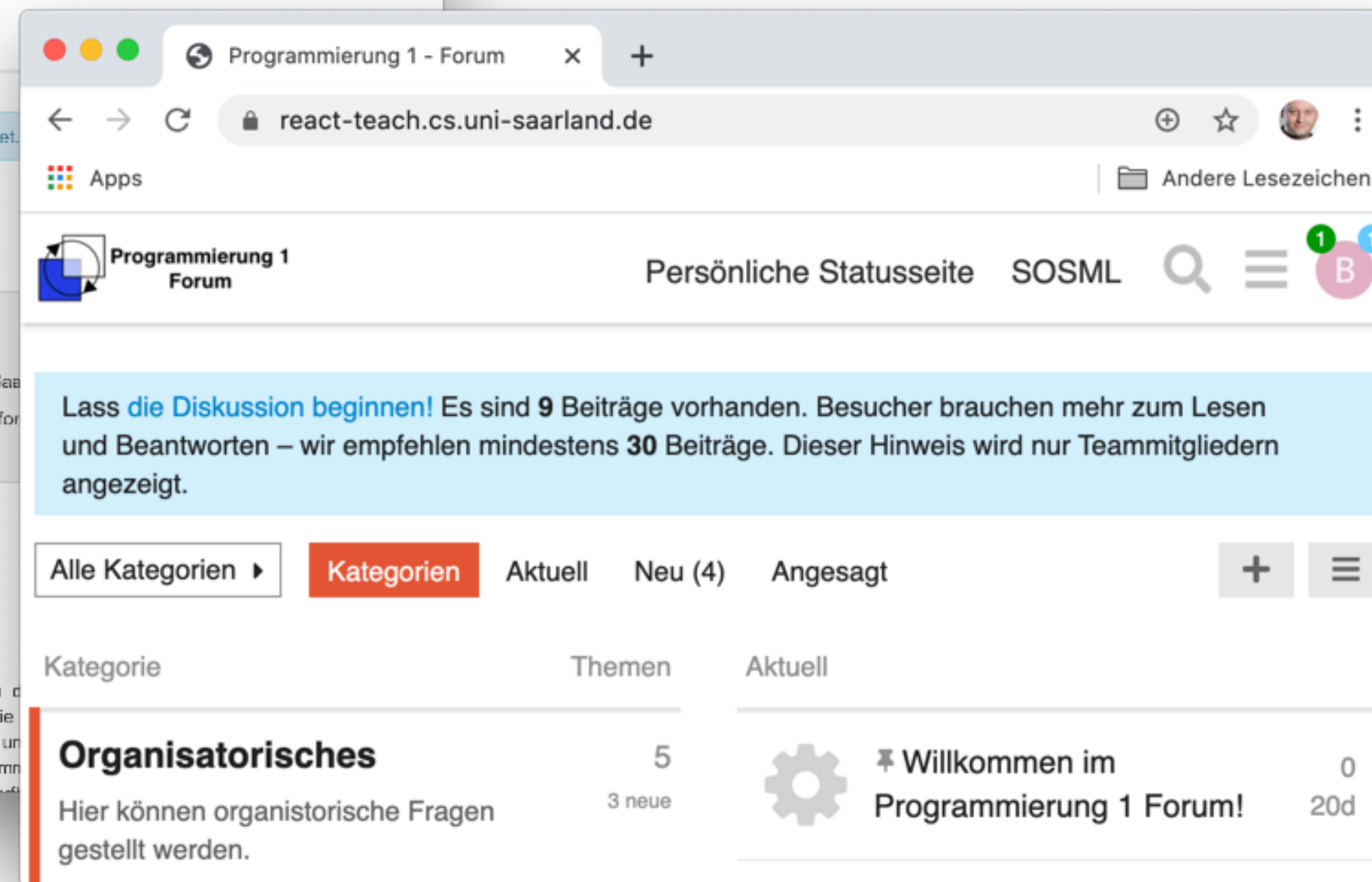
Not Okay

Speichern

www.prog1.saarland



www.prog1.saarland



forum.prog1.saarland

Tutorien

- ▶ Üben, üben, üben
 - ▶ Stoff der vergangenen Woche
 - ▶ Neue, spannende (und klausurrelevante) Aufgaben
 - ▶ Ihr Tutor/Ihre Tutorin steht Ihnen mit **Rat und Tat** zur Seite
-
- ▶ Beginn: nächste Woche
 - ▶ Tutorien finden auf Zoom und Discord statt.
 - ▶ Genaueres nächste Woche von Ihrem Tutor.

Übungsaufgaben

▶ **Übungsblatt**

- ▶ Jede Woche ein neues Übungsblatt (Dienstagabend im CMS)
- ▶ Übungsblatt wird nicht abgegeben
- ▶ Dient als Vorbereitung für den Minitest

▶ **Tutorienblatt**

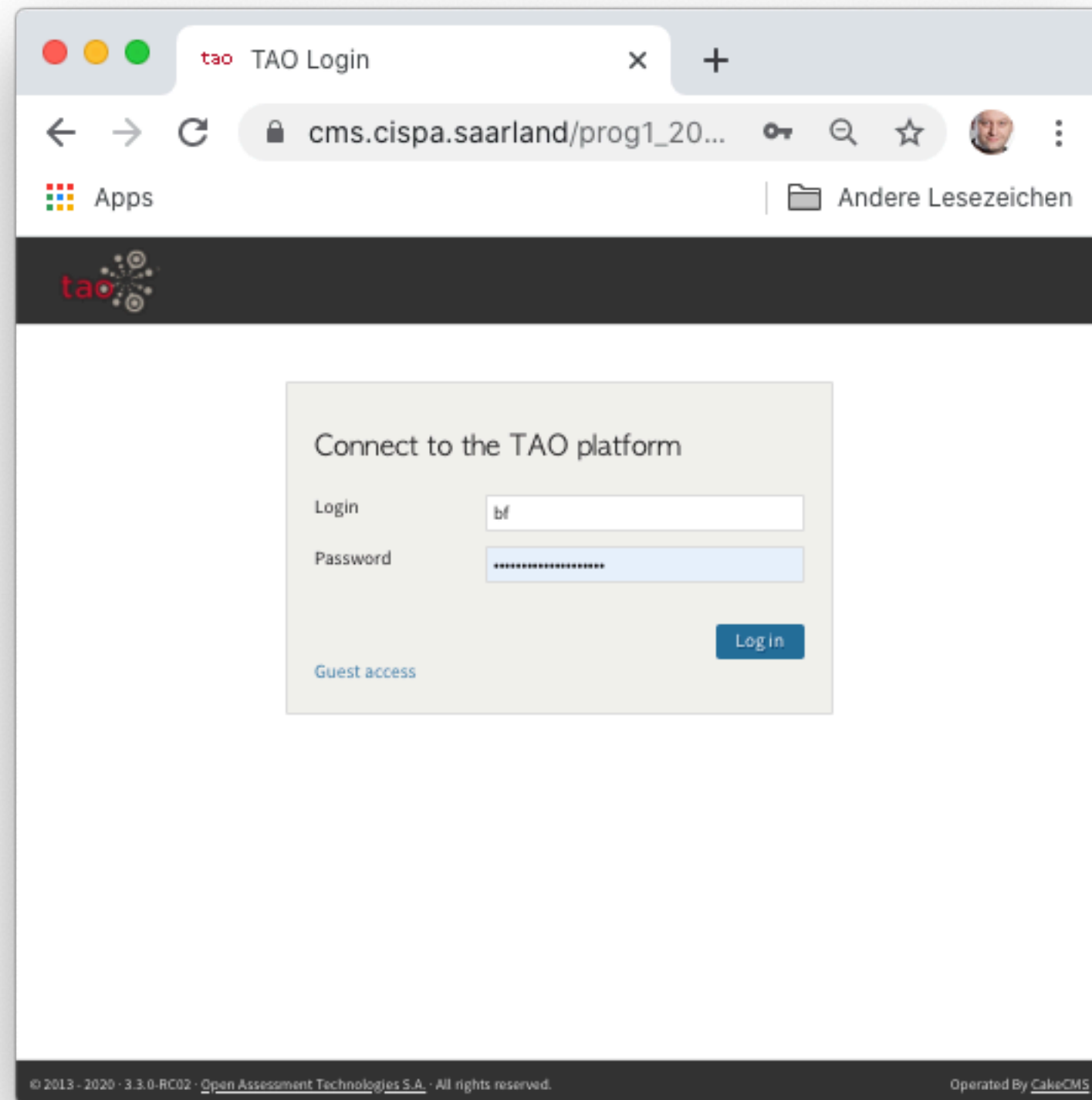
- ▶ Jede Woche ein neues Tutorienblatt (im Tutorium)
- ▶ Übungsblatt wird nicht abgegeben
- ▶ Dient zum Einüben und Festigen des Stoffes

Office Hours

- ▶ Gemeinsames Arbeiten an Übungsblättern
 - ▶ Hilfe und praktische Tipps von Ihren Tutoren
 - ▶ Effektive Vorbereitung auf Minitests und Klausuren
 - ▶ Wiederholung des Stoffs
-
- ▶ Mo 16:00-18:00, Di 12:00-14:00, Mi 12:00-14:00, Fr 14:00-16:00
 - ▶ ab morgen
 - ▶ Auf Discord
 - ▶ Geteilte Dateien via CodiMD



Minitests



minitest.prog1.saarland

- ▶ Stoff der vergangenen Woche
- ▶ jede Woche Do 10:00-10:15
- ▶ Login wie CMS
- ▶ Erster Minitest am 12. Nov
(beim ersten Mal nur Bonuspunkte)
- ▶ Reguläre Minitests ab 19. Nov

**50% der insgesamt erreichbaren Punkte nötig
um zur Hauptklausur zugelassen zu werden**

Klausuren

- ▶ **Mittelklausur 12.12.2020**

- ▶ Nachtermin: 9.1.2021
- ▶ Keine Zulassungsbeschränkung
- ▶ Geht zu **30%** in Ihre Note ein

- ▶ **Hauptklausur 13.02.2021**

- ▶ Nachtermin: 13.03.2021
- ▶ Zulassungsvoraussetzung: 50% der Minitestpunkte + bestandene Mittelklausur
- ▶ Geht zu **70%** in Ihre Note ein

Anmeldung zur Hauptklausur im LSF



Knobelpreis

- ▶ jede Woche eine **besonders herausfordernde** Aufgabe auf dem Übungsblatt
- ▶ gestellt und bewertet von Studierenden
- ▶ kann in Gruppen gelöst werden
- ▶ Gruppen mit **eigenem Teambatch** im Forum



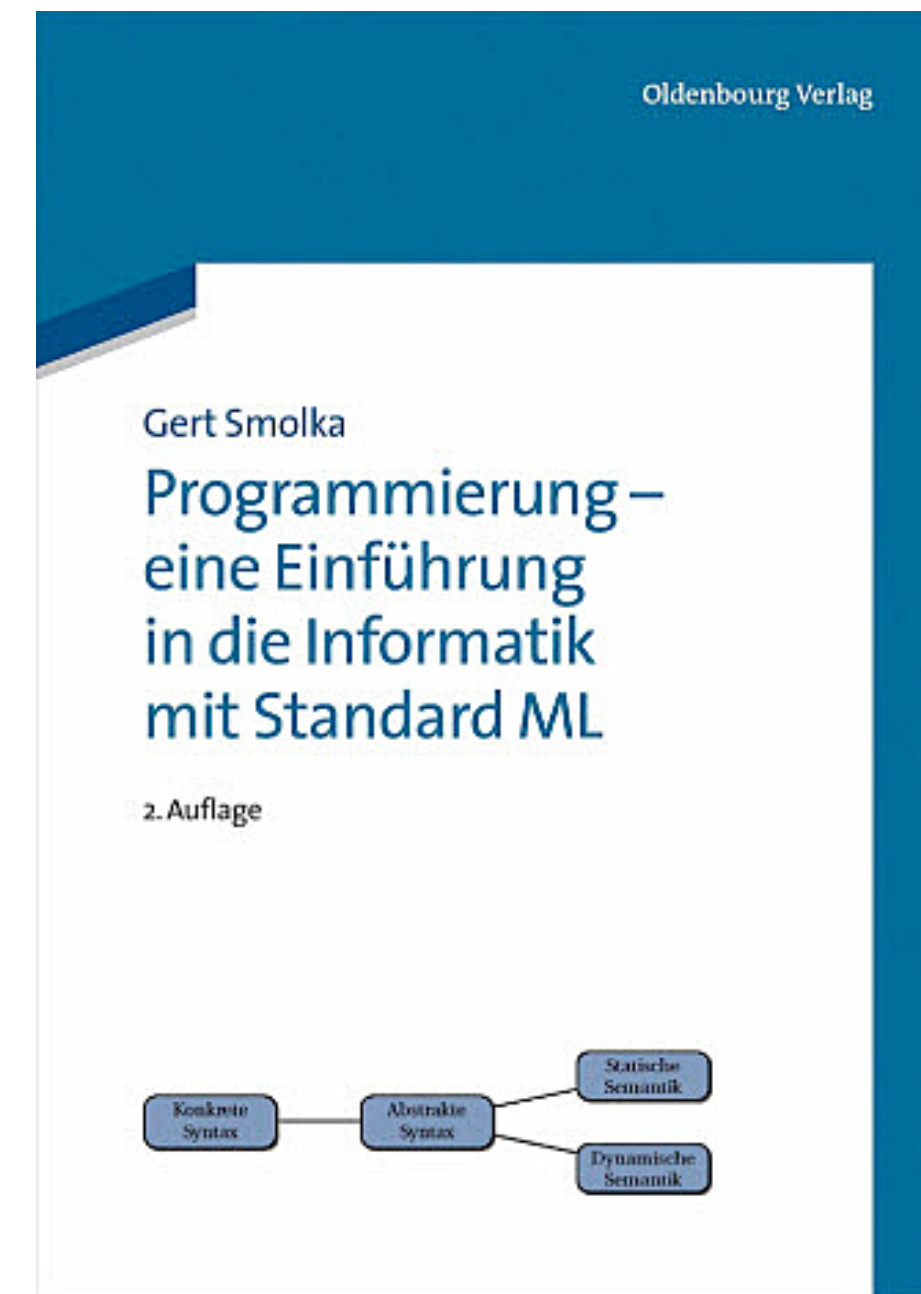
christopher.hahn  SML-Heroes

- ▶ Team-Anmeldung ab nächste Woche
- ▶ Lösung wird in einem extra Tutorium für Interessierte besprochen
- ▶ **Ruhm+Ehre+Preis für Gewinnergruppe**

Textbuch zur Vorlesung

Gert Smolka Programmierung — eine Einführung in die Informatik mit Standard ML

Die EBook-Ausgabe ist im
IP-Bereich der
Universität des Saarlandes
kostenfrei zugreifbar.



Programmierung 1

Das Ziel dieses Kurses ist, daß Sie lernen

- ▶ die **Grundkonzepte der Informatik** formal zu **beschreiben** und **anzuwenden**,
- ▶ **Programme** zu **erklären** und zu **analysieren**,
- ▶ strukturierte **Problemlösungen** zu **entwerfen**,
- ▶ die **Korrektheit** von Programmen zu **beweisen**,
- ▶ das Gelernte auf **neue Probleme anzuwenden**.

Es geht nicht darum, Wissen anzuhäufen,

sondern darum, Kompetenz

im **Beschreiben, Erklären, Analysieren,**

Argumentieren, Beweisen, und Anwenden

des Wissens zu erlangen.

Programmiersprache: Standard ML

- ▶ eine **funktionale** Programmiersprache
- ▶ minimalistisch
- ▶ übersichtlich
- ▶ einfache Syntax, klare Semantik
- ▶ ML Konzepte stecken in Clojure, Coq, Cyclone, C++, Elm, F#, F*, Haskell, Idris, Miranda, Nemerle, OCaml, Opa, Erlang, Rust, Scala

Ziel ist das **tiefe Verständnis**
der **programmiersprachlichen Konzepte**
so dass ein Einarbeiten in eine neue Sprache
innerhalb kürzester Zeit möglich ist.

1. Schnellkurs
2. Programmiersprachliches
3. Höherstufige Prozeduren
4. Listen und Strings
5. Sortieren
6. Konstruktoren und Ausnahmen
7. Bäume
8. Mengenlehre
9. Mathematische Prozeduren

10. Induktive Korrektheitsbeweise
11. Laufzeit rekursiver Programme
12. Statische und dynamische Semantik
13. Konkrete Syntax
14. Datenstrukturen
15. Speicher und veränderliche Objekte
16. Stapelmaschinen und Übersetzer



Kapitel 1

Schnellkurs

Unser erstes Programm

```
val x = 4*7+3
```

```
val y = x*(x-29)
```

- ▶ **Bezeichner:** x, y
dienen als Namen, die bei der Ausführung eines Programms an Werte gebunden werden.
- ▶ **Konstanten:** 3, 4, 7, 29
sind Wörter, die bestimmte Werte bezeichnen.
- ▶ **Operatoren:** +, -, *
sind Wörter, die Operationen darstellen.
- ▶ **Schlüsselwörter** val, =, (,)
dienen dazu, den Aufbau eines Programms darzustellen.

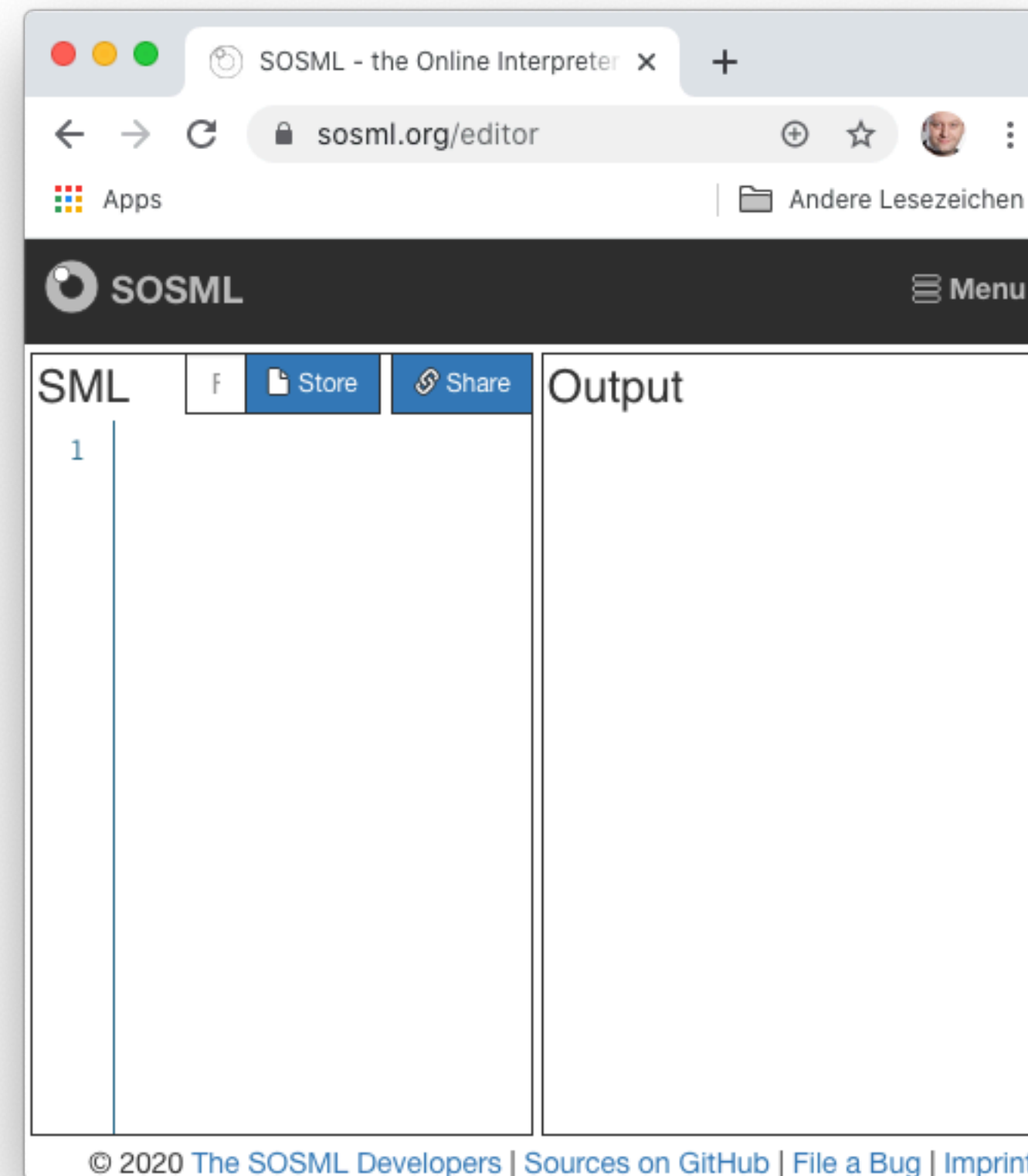
Deklarationen

val $\langle \textit{Bezeichner} \rangle = \langle \textit{Ausdruck} \rangle$

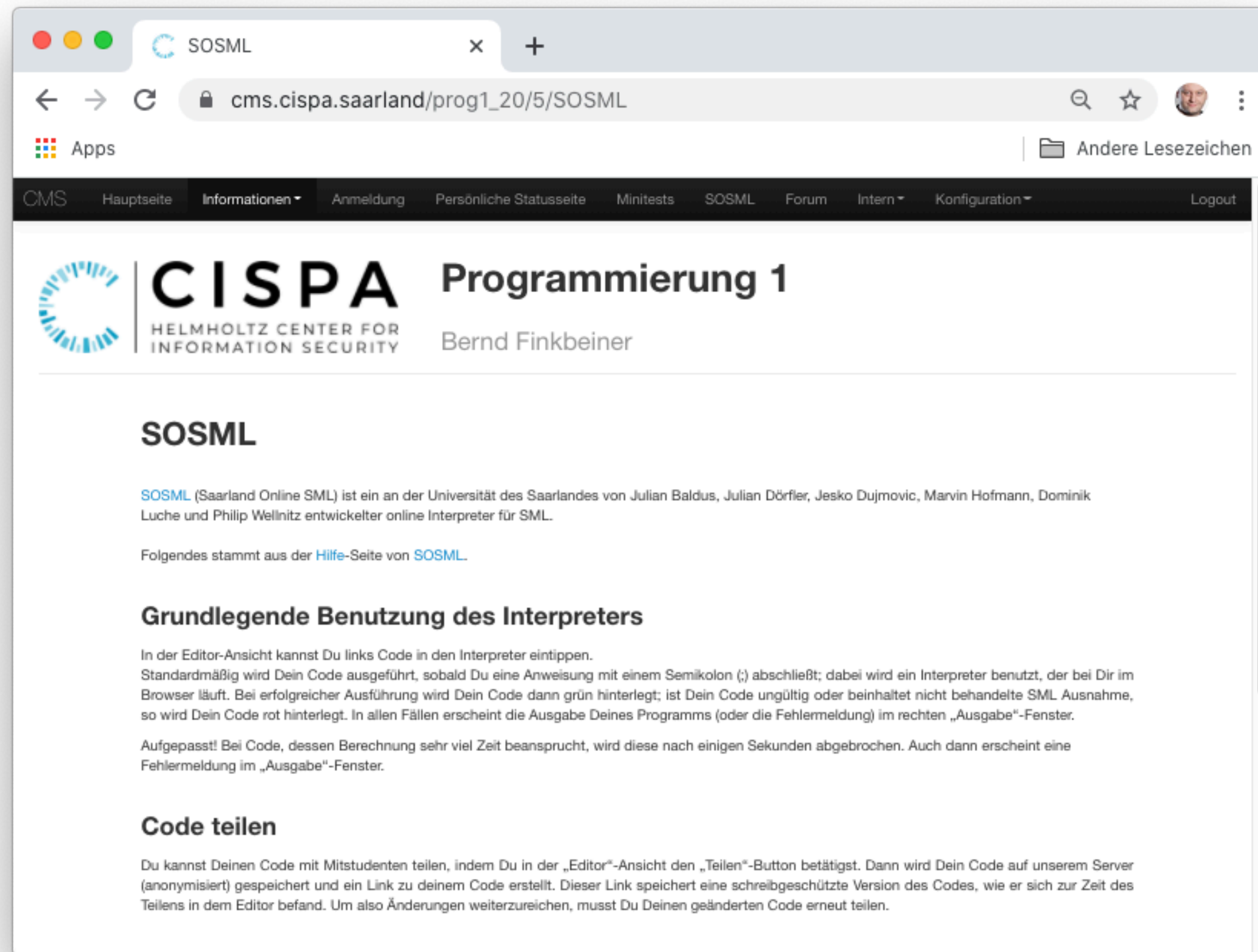
- ▶ **Ausdrücke** werden mit Konstanten, Operatoren, Bezeichnern und Klammern gebildet.
- ▶ Ein **Programm** ist eine Folge von Deklarationen.
- ▶ Bei der **Ausführung** eines Programms werden seine Deklarationen der Reihe nach ausgeführt.

Interpreter

- ▶ Ein **Interpreter** ist ein virtuelles Labor für das Experimentieren mit Programmen
- ▶ **SOSML** ist ein von Saarbrücker Informatikstudierenden entwickelter Interpreter für die Programmiersprache Standard ML.



sosml.org



www.prog1.saarland

Wir betrachten

Programme als implementierte Mathematik

Wir benutzen

eine **Programmiersprache** als **Labor**

um mit **implementierter Mathematik**
Experimente machen zu können.

- ▶ **Semikolon:** Damit der Interpreter mit der Bearbeitung des Programmes beginnt, muss ein Semikolon „;“ eingegeben werden.
- ▶ **Tilde „~“:** Negationsoperator für Zahlen
~2, ~1, 0, 1, 2
Aber: Subtraktionsoperator „-“: $x - 1$
- ▶ **Mehrfachdeklarationen:** es wird der durch die zuletzt ausgeführte Deklaration ermittelte Wert verwendet.
- ▶ **Ergebnisbezeichner it:** Deklarationen des Ergebnisbezeichners „it“ können ohne den Vorspann „val it =“ eingegeben werden.
- ▶ **Fehlermeldungen:** Der Interpreter prüft für jede Eingabe, ob sie gemäß den Regeln der Sprache zulässig ist.

Prozeduren

$$\textit{quadrat}(x) = x \cdot x$$

- ▶ Unter einer **Prozedur** verstehen wir eine Berechnungsvorschrift für eine Funktion.
- ▶ Prozeduren werden durch Gleichungen beschrieben.

- ▶ **Prozedurdeklaration:**

```
fun quadrat (x:int) = x*x
```

- ▶ **Ein und dieselbe Funktion kann durch verschiedene Prozeduren berechnet werden!**
- ▶ Eine Berechnungsvorschrift heißt **Algorithmus**.