Overview
CSC311: Theory of Computation

Reg Dodds
Department of Computer Science
University of the Western Cape
rdodds@uwc.ac.za

September 13, 2017

CSC 311 Theory of Computation
Second semester 2017

Theory of Computation

This course is an introduction to the theory of computation: the theory of *finite state automata* in their guises as regular languages, grammars, expressions and programs, and their applications in lexical analysers and *regex* machines; *context-free grammars* and their recognition and their applications in parsers; *Turing machines*; what they can do and what their limitations are; the halting problem; and some *decidability* results.

Framework of the course

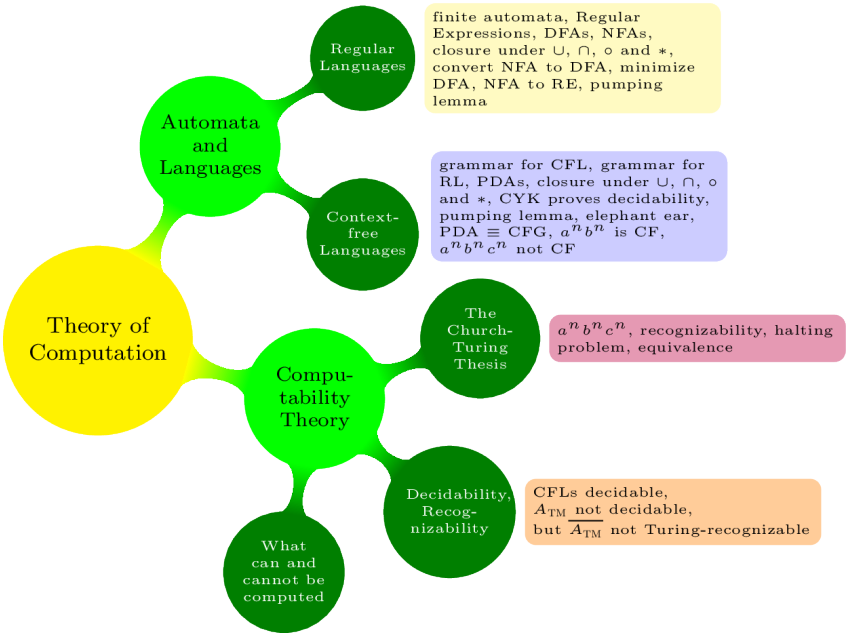

Figure 1: A mindmap for CSC311 Theory of Computation

DFAs, regular languages, regular expressions, regular grammars, NFAs, closure properties of RLs, converting an NFA into a DFA, the pumping lemma for RLs.

Context-free grammars, pushdown automata, converting between CFGs and PDAs, closure properties of CFLs, the pumping lemma for CFLs, non context-free languages.
Turing machines
Decidability, the halting problem, a TM unrecognizable language
Practical applications. We apply `flex` and apply `bison` to solve practical problems—this did not materialize in 2016

Theory of Computation:Outcomes

At the successful completion of this course the student has successfully implemented some DFAs, implemented NFA⇒DFA minimized a DFA, used `flex`, applied `bison` to grammars describing some small languages, programmed a generator for `first` and `follow` sets for a given grammar; solved some theoretical exercises; understood the halting problem; etc.
The student will cultivate a positive outlook and will develop the attitude that theory is useful because it can be applied to everyday problems in computing. The student will understand the power of finite state machines, context-free languages and Turing machines. The student will be conversant with tools that apply regular expressions and context-free languages to problems. The student will have emulated several machines.

Practicals and projects

Practical implementation of course work will be handed in weekly—every three weeks in 2017—Tutorial will also be conducted to exercise the theory.

Class material

All the transparencies for the course are available at the following link Four-per-A4 slide copies will be issued in class. Please don't print them yourself. The Latex for my own notes is available on request. You may download the complete manual for `make`, `flex` and `bison`. Since we are using a Sun system you may be compelled to use `lex` and `yacc` in the lab—this varies according to the whims of the Sun Lab staff..

Prescribed Book

Introduction to the Theory of Computation—1st, 2nd or 3rd Edition
by Michael Sipser
PWS Publishing Co., Thomson Course Technology, or Cengage Learning

C and C++ Books

The C Programming Language
by Brian Kernighan and Dennis M. Ritchie
Prentice Hall

C How to Program.
by Deitel and Deitel
Prentice Hall

C++ for Programmers—3rd Edition Leen Ammeraal
(2000) Wiley.

Other course material

The *class overheads* in A4 printable format are available as
The Theory of Computation slides
The slides are based loosely on Michael Sipser's book *Introduction to the Theory of Computation* and on Ken Louden's book on *Compiler Construction*
Please don't print the slides yourself—we supply them. It is essential that you read and study Sipser and do most of the exercises up to the end of Chapter 4 in order to come to grips with the course.

C for Java programmers slides

These slides are a quick introduction to C if you know Java and can be downloaded at the link
Slides for C for Java programmers

Regular Expressions

These notes by Reg Dodds are available at the link
Regular Expressions Notes

The notes are in a transient state but are a useful guide to using regular expressions in *flex*, `python` and `Java` and of course in *ToC*. Read these RE notes carefully. They are packed with useful hints, guidelines and examples.

For other helpful hints see my `regdodds` website at `csunx.uwc.ac.za`. All my up to date contact details can be found there.

Timetable CSC311: Theory of Computation

CSC 311: Theory of Computation timetable					
Period	**Monday**	**Tuesday**	**Wednesday**	**Thursday**	**Friday**
1 **08h30–09h00**					
2 **09h40–10h20**	CSC311 Computation Lecture SunLab or CS8				
3 **10h50–11h50**					
4 **12h00–13h00**					CSC311 Computation Lecture SunLab or CS8
Lunch **13h10–14h10**					
5 **14h20–15h20**					CSC311 Theory of Computation WinLab
6 **15h30–16h30**					CSC311 Theory of Computation WinLab
7 **16h40–17h40**					CSC311 Theory of Computation WinLab

Evalution of CSC311: Theory of Computation

The three parts of CSC311 are assessed independently, each counting one third of the total. The theory of computation is terminated with a final examination during the examination period. There class tests are arranged by chapter—when a chapter is completed a weighted test is given. Equal weights are assigned. Marks for all the tutorials and practicals together count thirty percent of the year mark.

Bibliography

Marvin Minsky. *Computation: Finite and Infinite Machines*, chapter 3. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1967.

Michael Sipser. *Introduction to the Theory of Computation*. Thomson Course Technology, second edition, 2006.

Michael Sipser. *Introduction to the Theory of Computation*. Cengage Learning, third edition, 2013.