Models of Computation
Revision Notes

James Brown

March 28, 2017

Contents

1 Introduction

These are notes I have written in preparation for the upcoming 2017 Models of Computation exam. This year the module was run by Paul Levy (P.B.Levy@cs.bham.ac.uk). This module is about problems and *computers*. We ask ourselves:

- What problems can be solved on a computer?
- What problems can be solved on a computer with finitely many states?
- What problems can be solved on a computer with only finitely many states, but also a stack of unlimited size?
- What problems can be solved on a computer with only finitely many states, but also a tape of unlimited size that it can read and write to?
- What problems can be solved *fast* on a computer?
- What does "fast" mean anyway?
- What does *computer* mean anyway?

2 Language Membership Problems and Regular Expressions

2.1 Language Membership Problems

Suppose we have a set of characters Σ, which we will call the *alphabet*.

2.2 Regex

Regular Expressions are a useful notation for describing languages.

3 Finite State Automata

4 Bisimulation and Minimisation

5 The Halting Problem

6 Properties of Code

7 Turing Machines

8 Church's Thesis

9 Complexity and P

10 NP

11 Lambda-calculus