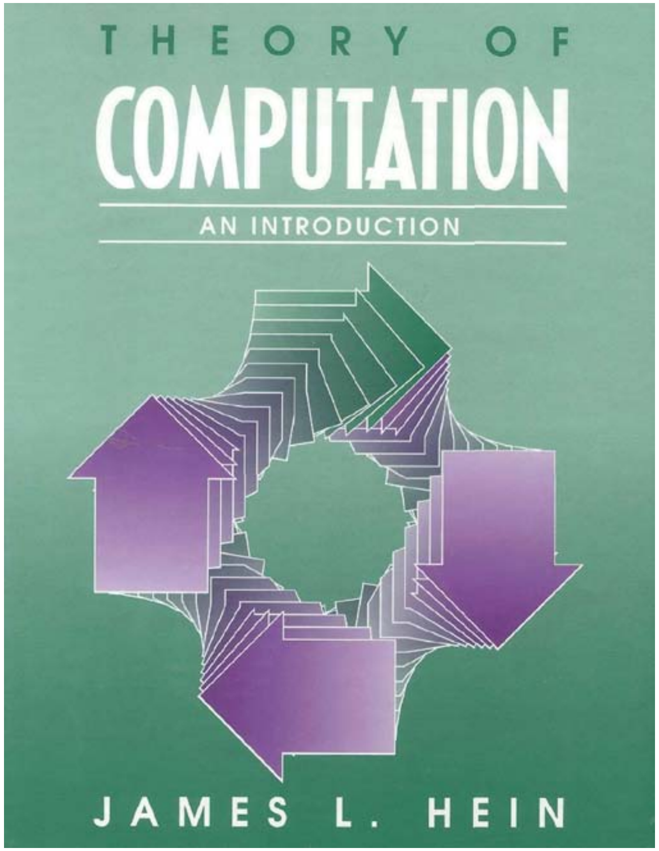
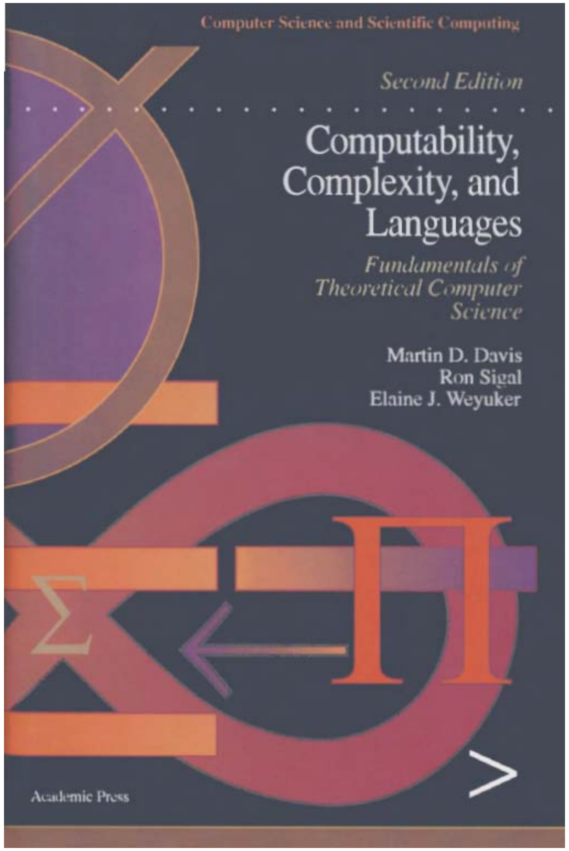
Theory of Computation

Lecture 01

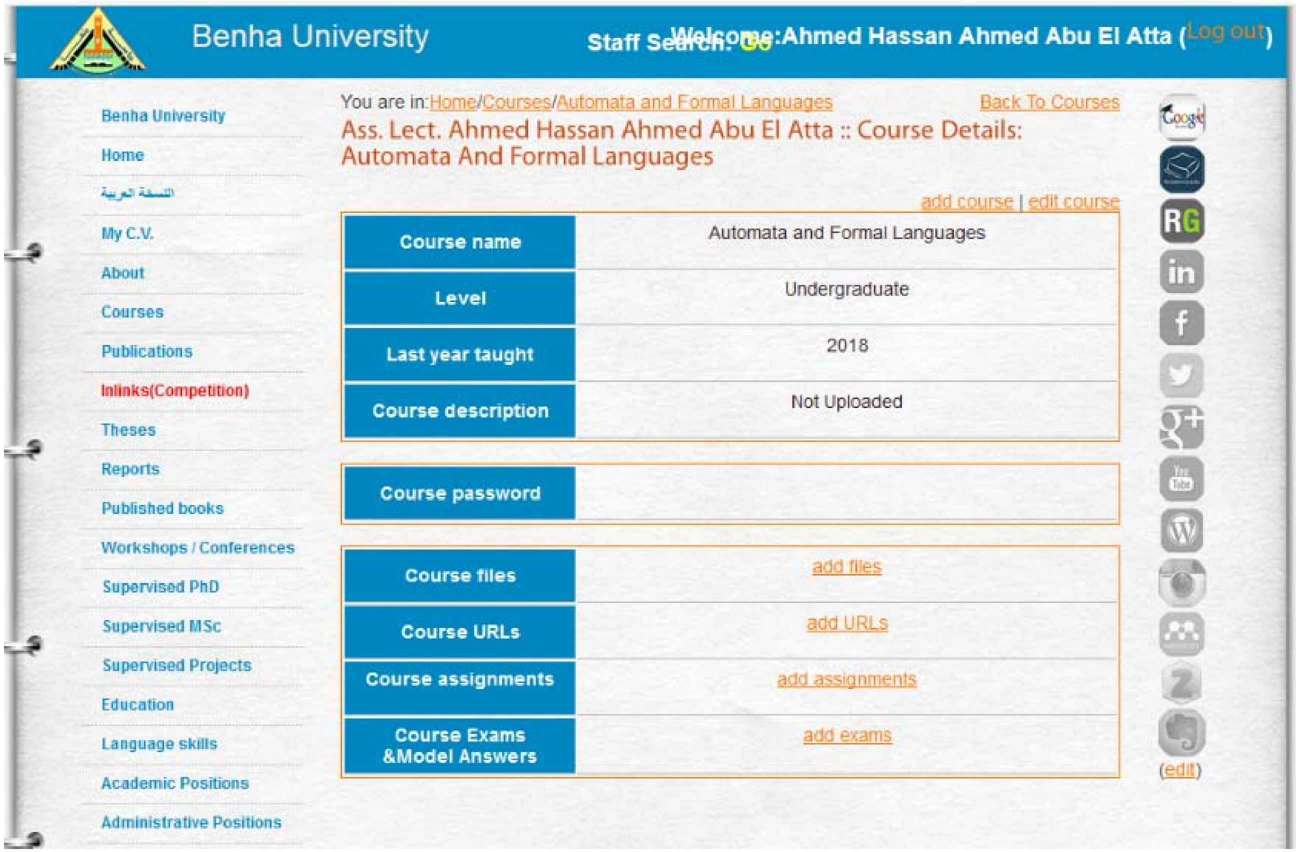
Books


THEORY OF
COMPUTATION
AN INTRODUCTION
JAMES L. HEIN


Computer Science and Scientific Computing
Second Edition
Computability, Complexity, and Languages
Fundamentals of Theoretical Computer Science
Martin D. Davis
Ron Sigal
Elaine J. Weyuker
Academic Press

PowerPoint

http://www.bu.edu.eg/staff/ahmedaboalatah14-courses/14767



Turing Machine

Agenda

- Turing machine
- Turing machine instruction
- EXAMPLE 1
- EXAMPLE 2
- Palindromes
- String Successor
- Find #

Turing machine

Turing machine is a computing device that was invented by the mathematician and logician Alan Turing (1912-1954).

The machine is described in the paper by Turing [1936].

It models the actions of a person doing a primitive calculation on a long strip of paper divided up into contiguous individual cells, each of which contains a symbol from a fixed alphabet.

Turing machine

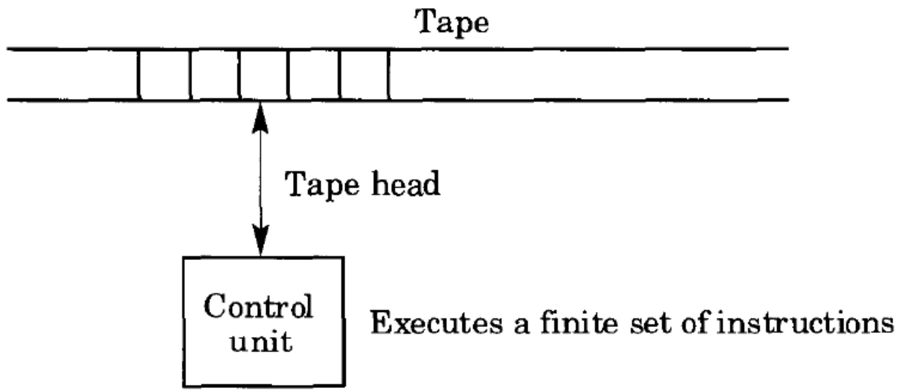
The person uses a pencil with an eraser.

Starting at some cell, the person observes the symbol in the cell and decides either to leave it alone or to erase it and write a new symbol in its place.

The person can then perform the same action on one of the adjacent cells.

Turing machine

- A Turing machine consists of two major components, a tape and a control unit.
- The *tape is a sequence of cells that extends to infinity in* both directions.
- Each cell contains a symbol from a finite alphabet.
- There is a tape head that reads from a cell and writes into the same cell.
- The *control* unit contains a finite set of instructions, which are executed as follows:
  - Each instruction causes the tape head to read the symbol from a cell,
  - to write a symbol into the same cell, and
  - either to move the tape head to an adjacent cell or to leave it at the same cell.

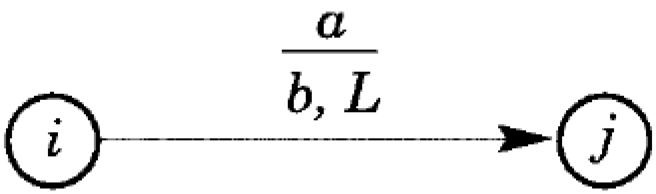


Turing machine instruction

Each Turing machine instruction contains the following five parts:

- The current machine state.
- A tape symbol read from the current tape cell.
- A tape symbol to write into the current tape cell.
- A direction for the tape head to move.
- The next machine state.

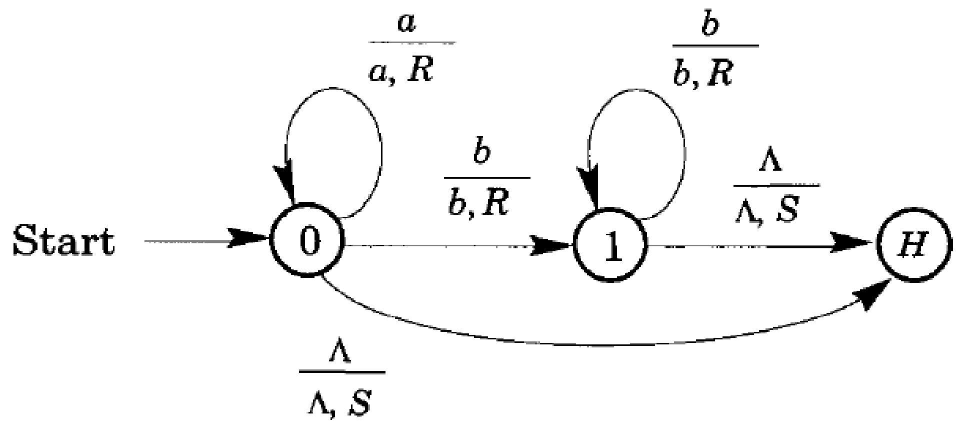
$$\langle i, a, b, L, j \rangle$$



EXAMPLE 1

EXAMPLE 1. Suppose we want to write a Turing machine to recognize the language $\{a^n b^m \mid m, n \in \mathbb{N}\}$. Of course this is a regular language, represented by the regular expression a^*b^*. So there is a DFA to recognize it. Of course there is also a PDA to recognize it. So there had better be a Turing machine to recognize it.

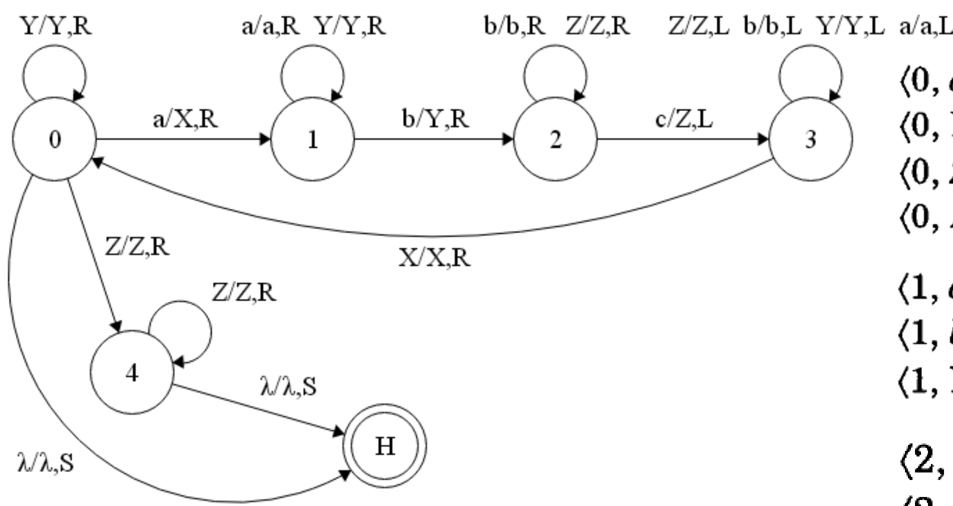
$\langle 0, \Lambda, \Lambda, S, \text{Halt}\rangle$	The string is Λ.
$\langle 0, a, a, R, 0\rangle$	Scan a's.
$\langle 0, b, b, R, 1\rangle$	
$\langle 1, b, b, R, 1\rangle$	Scan b's.
$\langle 1, \Lambda, \Lambda, S, \text{Halt}\rangle$	



EXAMPLE 2

EXAMPLE 2. To show the power of Turing machines, we'll construct a Turing machine to recognize the following language:

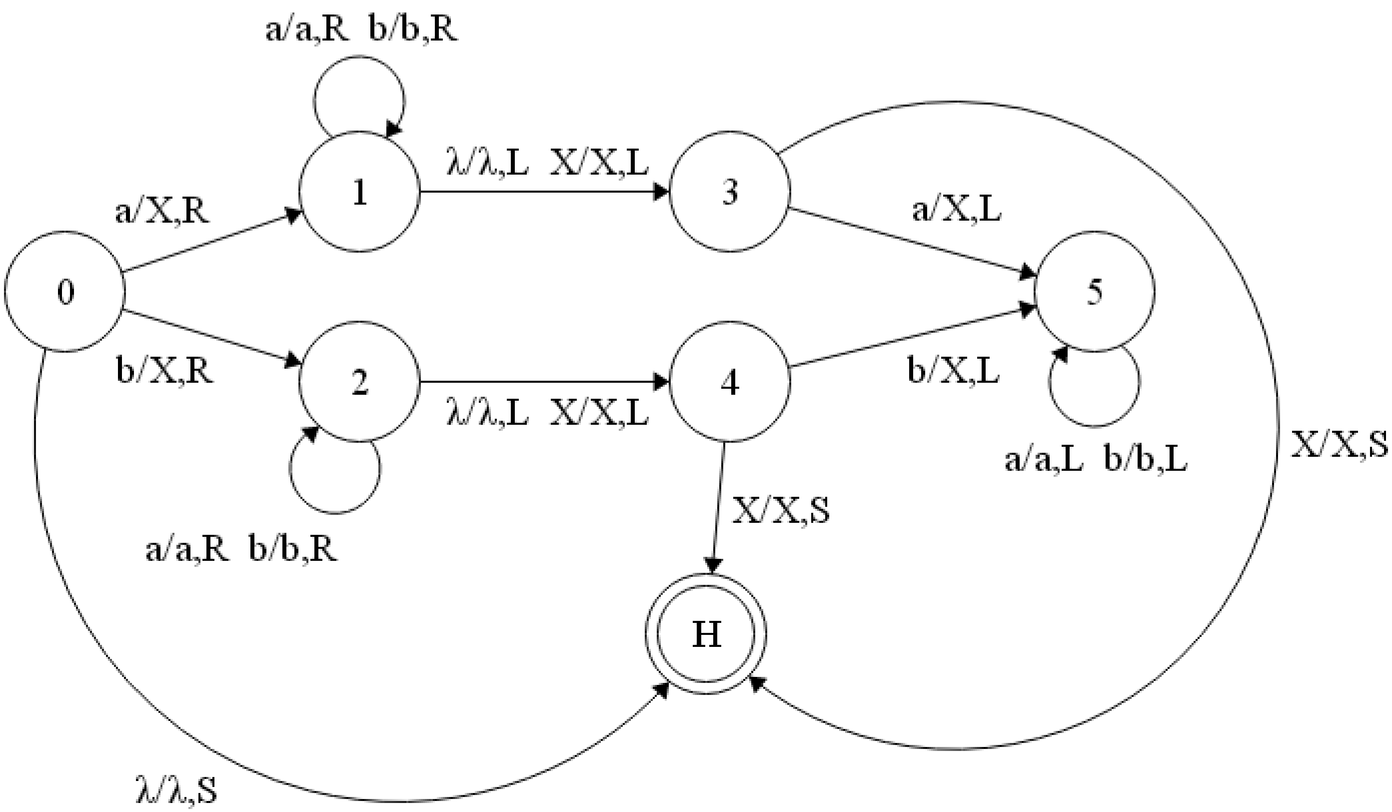
$$\{a^n b^n c^n \mid n \geq 0\}.$$



$\langle 0, a, X, R, 1\rangle$	Replace a by X and scan right.
$\langle 0, Y, Y, R, 0\rangle$	Scan right.
$\langle 0, Z, Z, R, 4\rangle$	Go make the final check.
$\langle 0, \Lambda, \Lambda, S, \text{Halt}\rangle$	Success.
$\langle 1, a, a, R, 1\rangle$	Scan right.
$\langle 1, b, Y, R, 2\rangle$	Replace b by Y and scan right.
$\langle 1, Y, Y, R, 1\rangle$	Scan right.
$\langle 2, c, Z, L, 3\rangle$	Replace c by Z and scan left.
$\langle 2, b, b, R, 2\rangle$	Scan right.
$\langle 2, Z, Z, R, 2\rangle$	Scan right.
$\langle 3, a, a, L, 3\rangle$	Scan left.
$\langle 3, b, b, L, 3\rangle$	Scan left.
$\langle 3, X, X, R, 0\rangle$	Found X. Move right one cell.
$\langle 3, Y, Y, L, 3\rangle$	Scan left.
$\langle 3, Z, Z, L, 3\rangle$	Scan left.
$\langle 4, Z, Z, R, 4\rangle$	Scan right.
$\langle 4, \Lambda, \Lambda, S, \text{Halt}\rangle$	Success.

Palindromes

Construct a Turing machine to recognize the language of all palindromes over {a, b}.

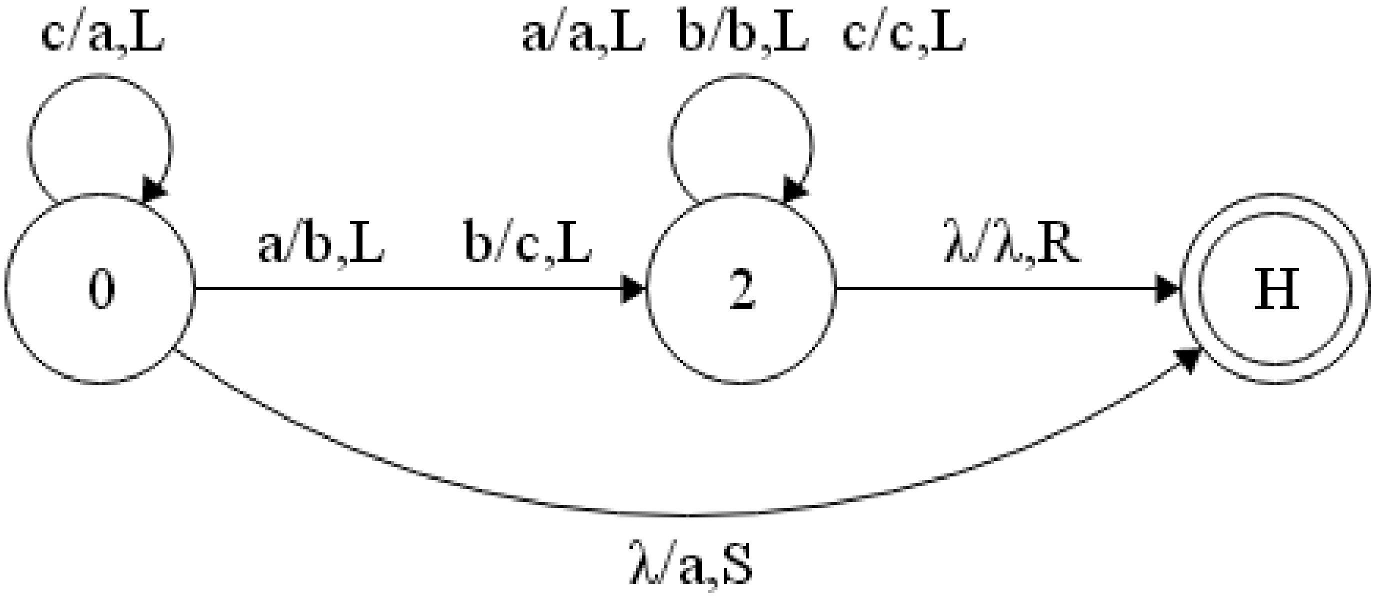


String Successor

Construct a single-tape Turing machine that inputs any string over the alphabet {a, b, c} *and outputs its successor in the standard ordering,* where we assume that a ***< b < c.***

Recall that in the standard ordering, strings are ordered by length, strings of the same length being ordered lexicographically.

String Successor



Find

Construct a Turing machine that starts with the symbol # in one cell, where all other tape cells are blank. The beginning position of the tape head is not known. The machine should halt with the tape head pointing at the cell containing #, all other tape cells being blank.