CS 301 - Lecture 20
Combining Turing Machines and Turing's Thesis

Fall 2008

Review

- Languages and Grammars
  - Alphabets, strings, languages
- Regular Languages
  - Deterministic Finite and Nondeterministic Automata
  - Equivalence of NFA and DFA
  - Regular Expressions
  - Regular Grammars
  - Properties of Regular Languages
  - Languages that are not regular and the pumping lemma
- Context Free Languages
  - Context Free Grammars
  - Derivations: leftmost, rightmost and derivation trees
  - Parsing and ambiguity
  - Simplifications and Normal Forms
  - Nondeterministic Pushdown Automata
  - Pushdown Automata and Context Free Grammars
  - Deterministic Pushdown Automata
  - Pumping Lemma for context free grammars
  - Properties of Context Free Grammars
- Turing Machines
  - Definition and Accepting Languages
  - Today: Computing Functions, Combining Machines, and Turing's Thesis

Standard Turing Machine

The machine we described is the standard:

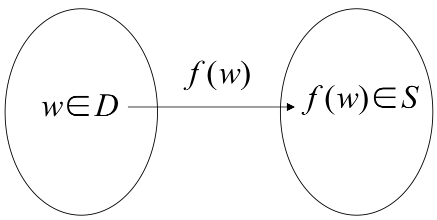
- Deterministic
- Infinite tape in both directions
- Tape is the input/output file

Computing Functions with Turing Machines

A function $f(w)$ has:

Domain: D

Result Region: S



A function may have many parameters:

Example: Addition function

$$f(x, y) = x + y$$

Integer Domain

Decimal: 5

Binary: 101

Unary: 11111

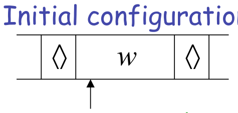
We prefer **unary** representation:

easier to manipulate with Turing machines

Definition:

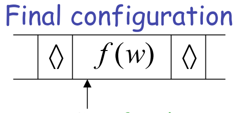
A function f is computable if there is a Turing Machine M such that:

Initial configuration



q_0 initial state

Final configuration



q_f final state

For all $w \in D$ Domain

In other words:

A function f is computable if
there is a Turing Machine M such that:

$$q_0\, w \overset{*}{\succ} q_f\, f(w)$$

Initial Configuration

Final Configuration

For all $w \in D$ Domain

Example

The function $f(x, y) = x + y$ is computable

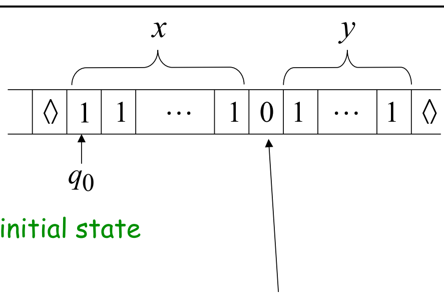
x, y are integers

Turing Machine:

Input string: $x0y$ unary

Output string: $xy0$ unary

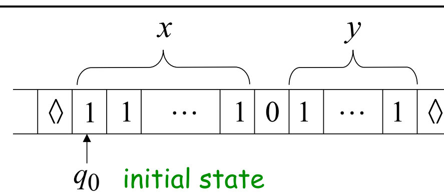
Start



initial state

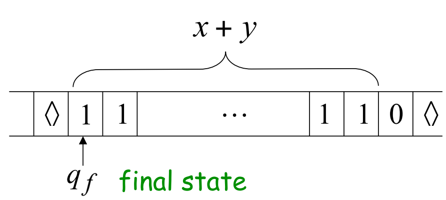
The 0 is the delimiter that separates the two numbers

Start



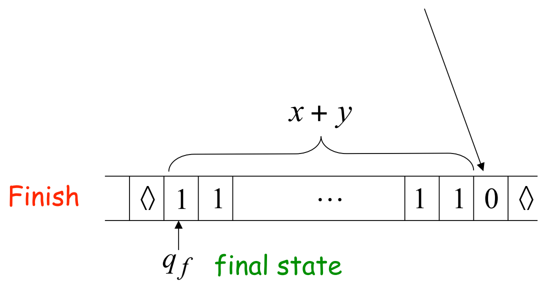
initial state

Finish

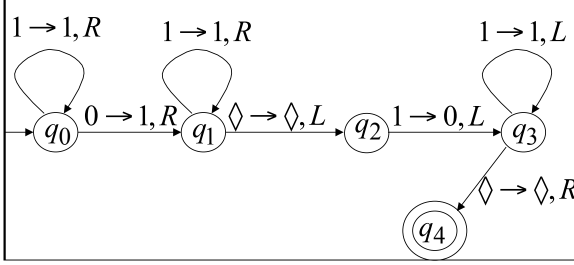


final state

The 0 helps when we use
the result for other operations



Turing machine for function $f(x,y)=x+y$

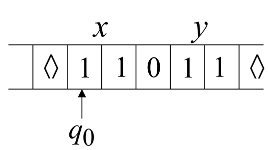


Execution Example:

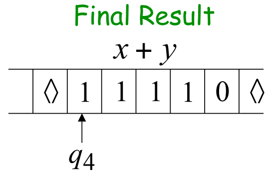
$x = 11$ (2)

$y = 11$ (2)

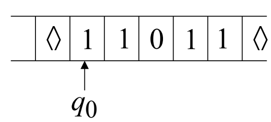
Time 0

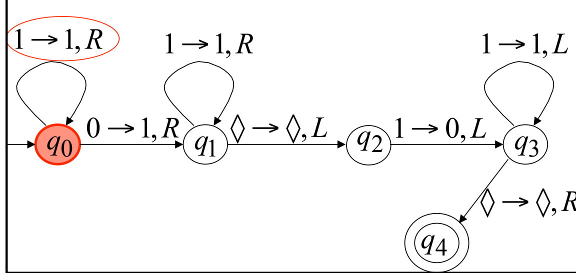


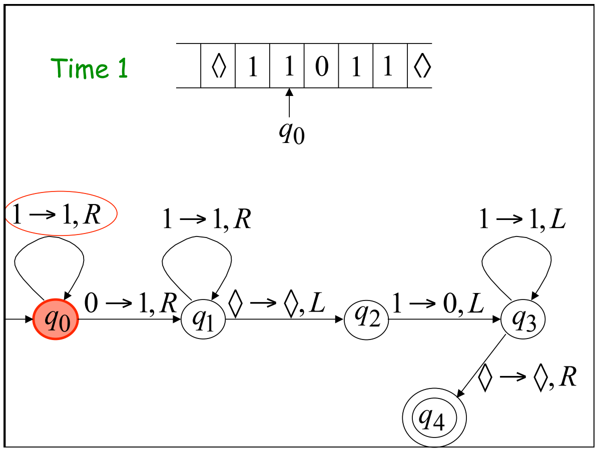
Final Result

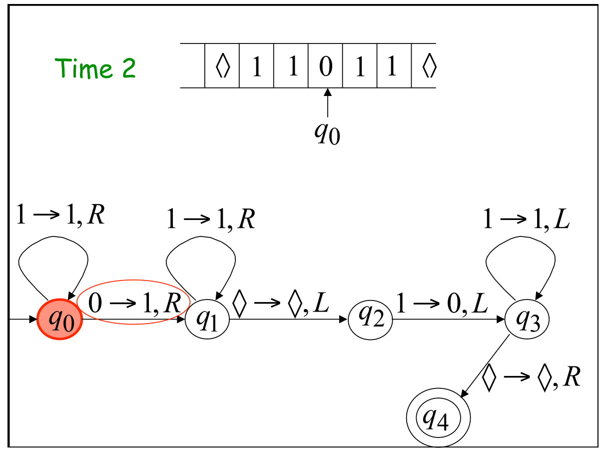


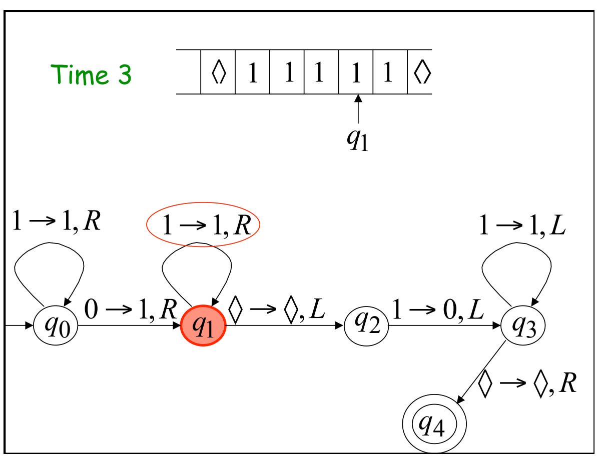
Time 0

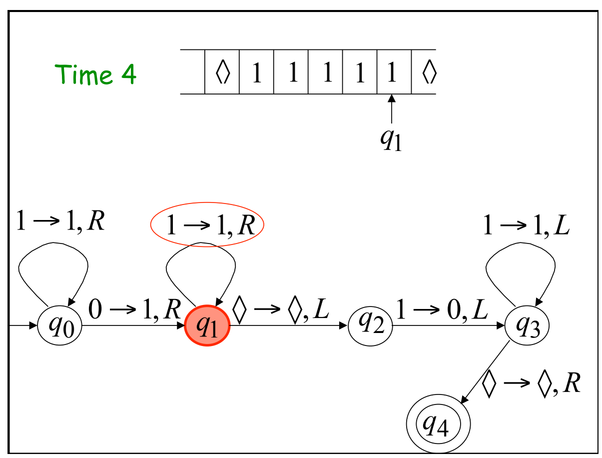


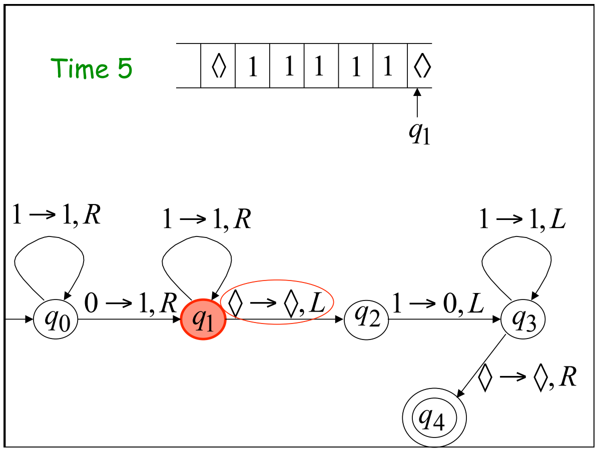


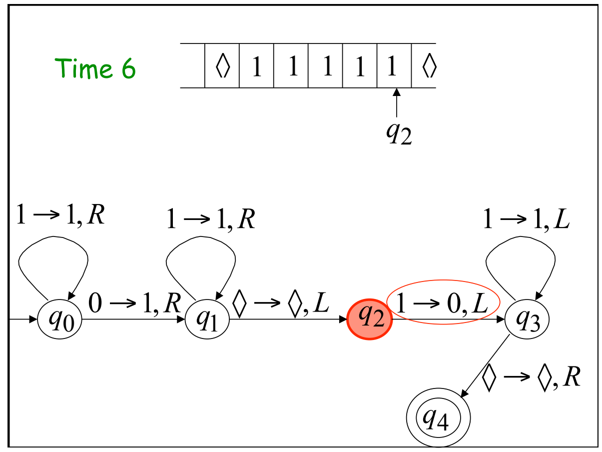

Time 1
◊ 1 1 0 1 1 ◊
q_0
$1 \to 1, R$
$0 \to 1, R$
$1 \to 1, R$
$◊ \to ◊, L$
$1 \to 0, L$
$1 \to 1, L$
$◊ \to ◊, R$
q_0 q_1 q_2 q_3 q_4

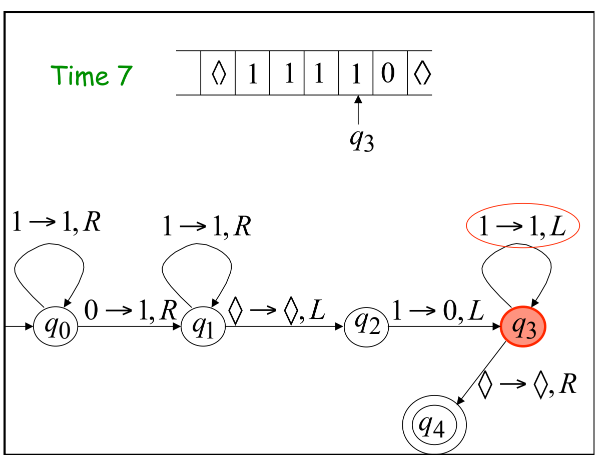

Time 2
◊ 1 1 0 1 1 ◊
q_0
$1 \to 1, R$
$0 \to 1, R$
$1 \to 1, R$
$◊ \to ◊, L$
$1 \to 0, L$
$1 \to 1, L$
$◊ \to ◊, R$
q_0 q_1 q_2 q_3 q_4

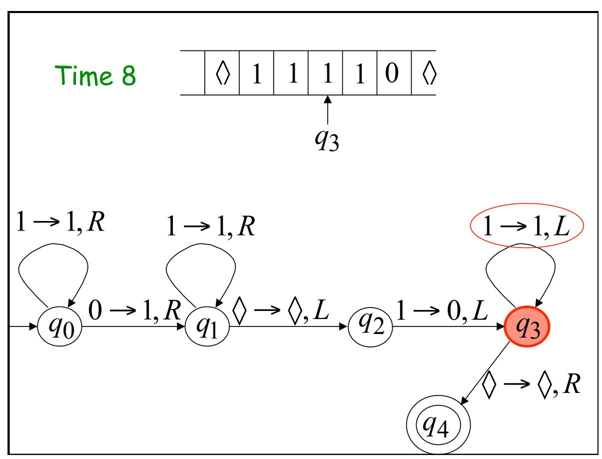

Time 3
◊ 1 1 1 1 1 ◊
q_1
$1 \to 1, R$
$0 \to 1, R$
$1 \to 1, R$
$◊ \to ◊, L$
$1 \to 0, L$
$1 \to 1, L$
$◊ \to ◊, R$
q_0 q_1 q_2 q_3 q_4

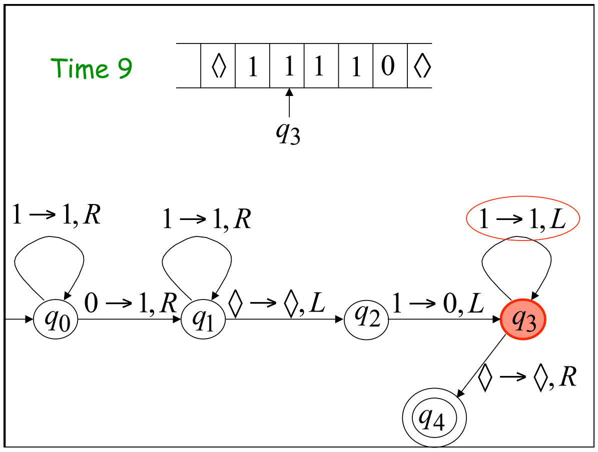

Time 4
◊ 1 1 1 1 1 ◊
q_1
$1 \to 1, R$
$0 \to 1, R$
$1 \to 1, R$
$◊ \to ◊, L$
$1 \to 0, L$
$1 \to 1, L$
$◊ \to ◊, R$
q_0 q_1 q_2 q_3 q_4

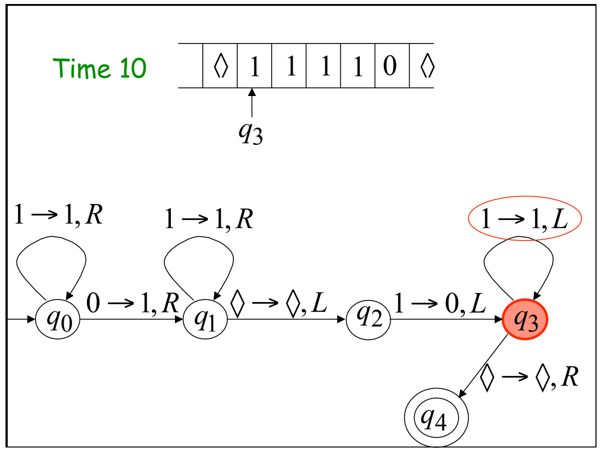

Time 5
◊ 1 1 1 1 1 ◊
q1
1 → 1, R
1 → 1, R
1 → 1, L
q0
0 → 1, R
q1
◊ → ◊, L
q2
1 → 0, L
q3
◊ → ◊, R
q4

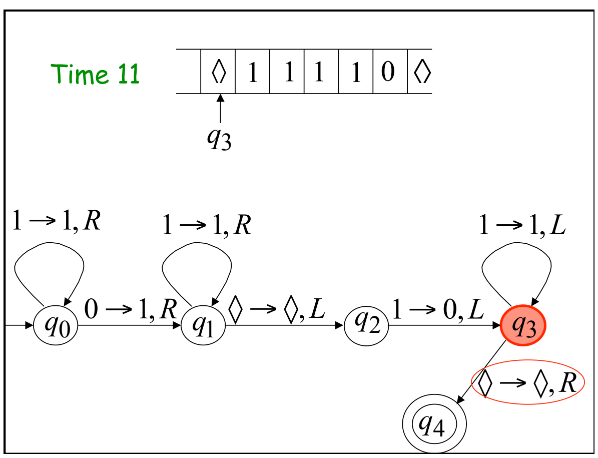

Time 6
◊ 1 1 1 1 1 ◊
q2
1 → 1, R
1 → 1, R
1 → 1, L
q0
0 → 1, R
q1
◊ → ◊, L
q2
1 → 0, L
q3
◊ → ◊, R
q4

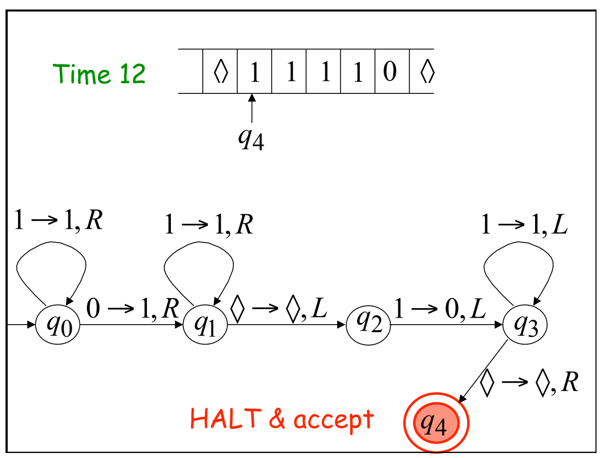

Time 7
◊ 1 1 1 1 0 ◊
q3
1 → 1, R
1 → 1, R
1 → 1, L
q0
0 → 1, R
q1
◊ → ◊, L
q2
1 → 0, L
q3
◊ → ◊, R
q4


Time 8
◊ 1 1 1 1 0 ◊
q3
1 → 1, R
1 → 1, R
1 → 1, L
q0
0 → 1, R
q1
◊ → ◊, L
q2
1 → 0, L
q3
◊ → ◊, R
q4


Time 9
◊ 1 1 1 1 0 ◊
q_3
$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
q_0
$0 \to 1, R$
q_1
$◊ \to ◊, L$
q_2
$1 \to 0, L$
q_3
$◊ \to ◊, R$
q_4


Time 10
◊ 1 1 1 1 0 ◊
q_3
$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
q_0
$0 \to 1, R$
q_1
$◊ \to ◊, L$
q_2
$1 \to 0, L$
q_3
$◊ \to ◊, R$
q_4


Time 11
◊ 1 1 1 1 0 ◊
q_3
$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
q_0
$0 \to 1, R$
q_1
$◊ \to ◊, L$
q_2
$1 \to 0, L$
q_3
$◊ \to ◊, R$
q_4


Time 12
◊ 1 1 1 1 0 ◊
q_4
$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
q_0
$0 \to 1, R$
q_1
$◊ \to ◊, L$
q_2
$1 \to 0, L$
q_3
$◊ \to ◊, R$
HALT & accept
q_4

Another Example

The function $f(x) = 2x$ is computable

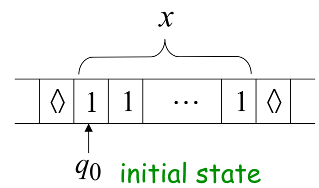
x is integer

Turing Machine:

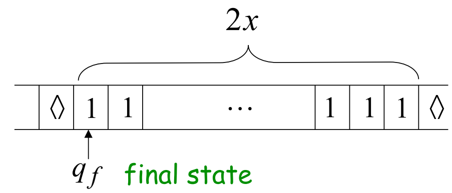
Input string: x unary

Output string: xx unary

Start



Finish

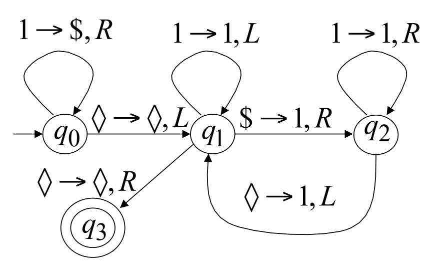


Turing Machine Pseudocode for $f(x) = 2x$

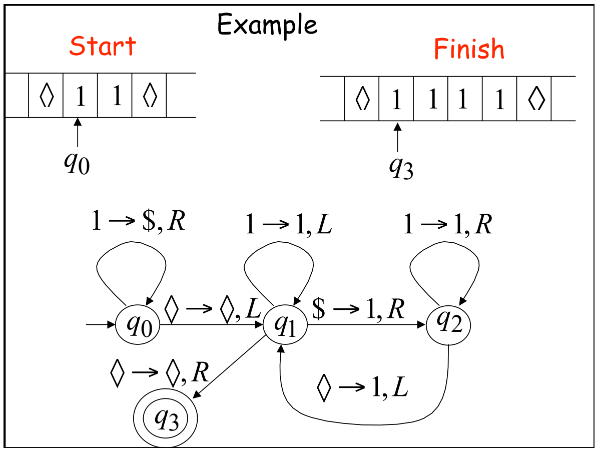
- Replace every 1 with \$
- Repeat:
  - Find rightmost \$, replace it with 1
  - Go to right end, insert 1

Until no more \$ remain

Turing Machine for $f(x) = 2x$



Example



Another Example

The function

is computable

$$f(x,y) = \begin{cases} 1 & \text{if } x > y \\ 0 & \text{if } x \leq y \end{cases}$$

Turing Machine for

$$f(x,y) = \begin{cases} 1 & \text{if } x > y \\ 0 & \text{if } x \leq y \end{cases}$$

Input: $x0y$

Output: 1 or 0

Turing Machine Pseudocode:

- Repeat

  Match a 1 from x with a 1 from y

  Until all of x or y is matched

- If a 1 from x is not matched

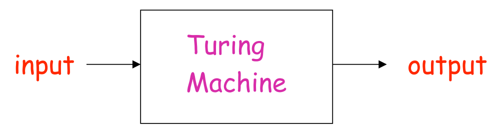
  erase tape, write 1 $(x > y)$

  else

  erase tape, write 0 $(x \leq y)$

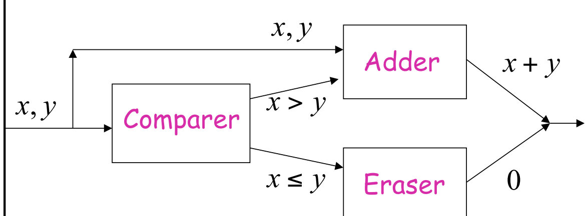
Combining Turing Machines

Block Diagram



Example:

$$f(x,y) = \begin{cases} x+y & \text{if } x > y \\ 0 & \text{if } x \le y \end{cases}$$



Turing's Thesis

Turing's thesis:

Any computation carried out
by mechanical means
can be performed by a Turing Machine

(1930)

Computer Science Law:

A computation is mechanical
if and only if
it can be performed by a Turing Machine

There is no known model of computation
more powerful than Turing Machines

Definition of Algorithm:

An algorithm for function $f(w)$
is a
Turing Machine which computes $f(w)$

Algorithms are Turing Machines

When we say:

There exists an algorithm

We mean:

There exists a Turing Machine
that executes the algorithm

What's Next

- Read
  - Linz Chapter 1,2.1, 2.2, 2.3, (skip 2.4), 3, 4, 5, 6.1, 6.2, (skip 6.3), 7.1, 7.2, 7.3, (skip 7.4), 8, 9, 10.1, 10.2, and 10.3
  - JFLAP Chapter 1, 2.1, (skip 2.2), 3, 4, 5, 6, 7, (skip 8), 9
- Next Lecture Topics From 10.1, 10.2 and 10.3
  - Turing Machine Variations
- Quiz 3 in Recitation on Wednesday 11/12
  - Covers Linz 7.1, 7.2, 7.3, (skip 7.4), 8, and JFLAP 5,6,7
  - Closed book, but you may bring one sheet of 8.5 x 11 inch paper with any notes you like.
  - Quiz will take the full hour
- Homework
  - Homework Due Today
  - New Homework Available Friday Morning
  - New Homework Due Next Thursday