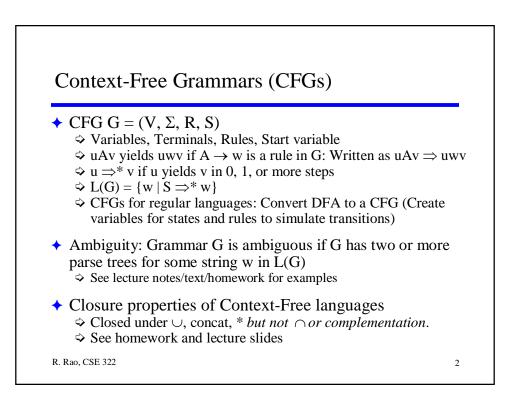
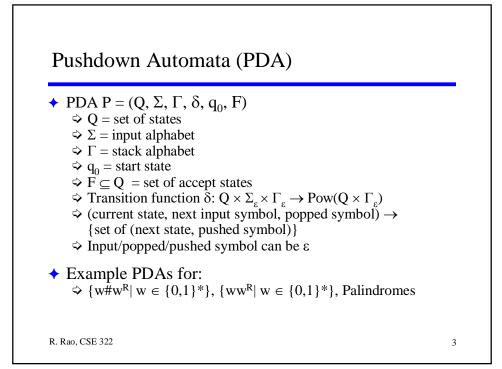


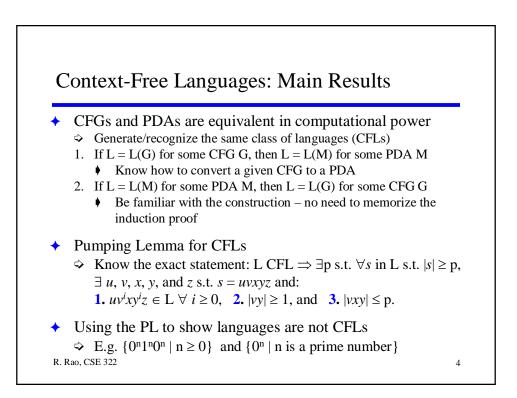
- See Midterm Review Slides
 - \Rightarrow Emphasis on:
 - Sets, strings, and languages
 - Operations on strings/languages (concat, *, union, etc)
 - Lexicographic ordering of strings
 - DFAs and NFAs: definitions and how they work
 - Regular languages and properties
 - Regular expressions and GNFAs (see lecture slides)
 - Pumping lemma for regular languages and showing nonregularity

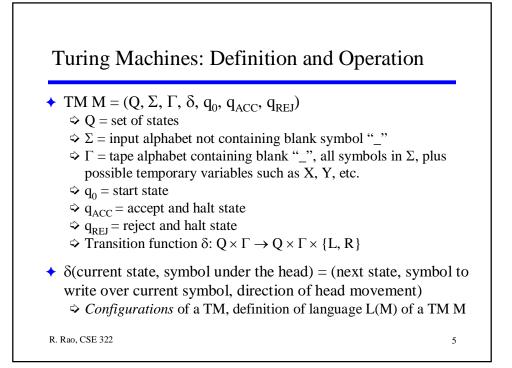
1

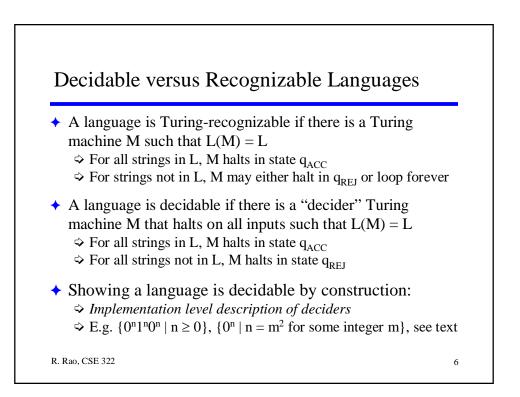
R. Rao, CSE 322

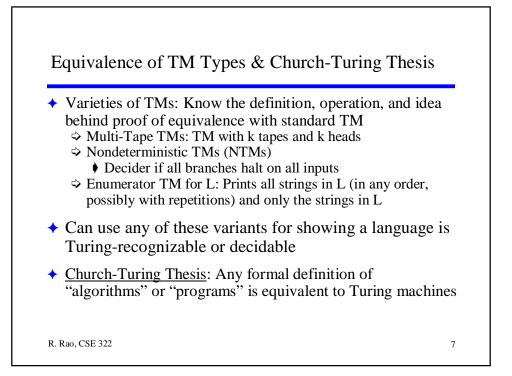


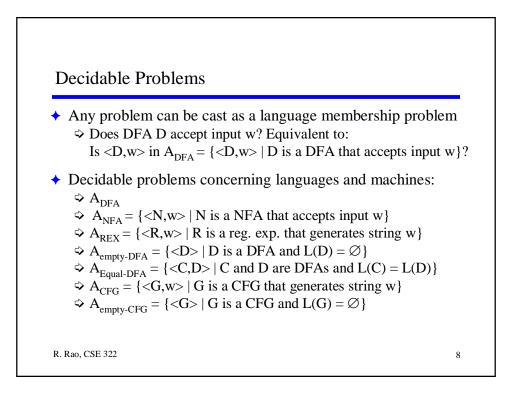


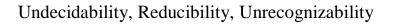












- ◆ To show a problem A is undecidable, reduce A_{TM} to A
 ⇒ Show that if A was decidable, then you can use the decider for A as a *subroutine* to decide A_{TM}
 - E.g. Halting problem = "Does a program halt for an input or go into an infinite loop?"
 - $\label{eq:analytical_constraint} \begin{array}{l} \diamondsuit \\ \text{Can show that the Halting problem is undecidable by reducing} \\ A_{\text{TM}} \text{ to } A_{\text{H}} = \{ \ <\!M,\!w\!> \mid TM \ M \ \text{halts on input } w \} \end{array}$

• A is decidable iff A and \overline{A} are both Turing-recognizable

 \Rightarrow Corollary: \overline{A}_{TM} and \overline{A}_{H} are not Turing-recognizable

R. Rao, CSE 322

9