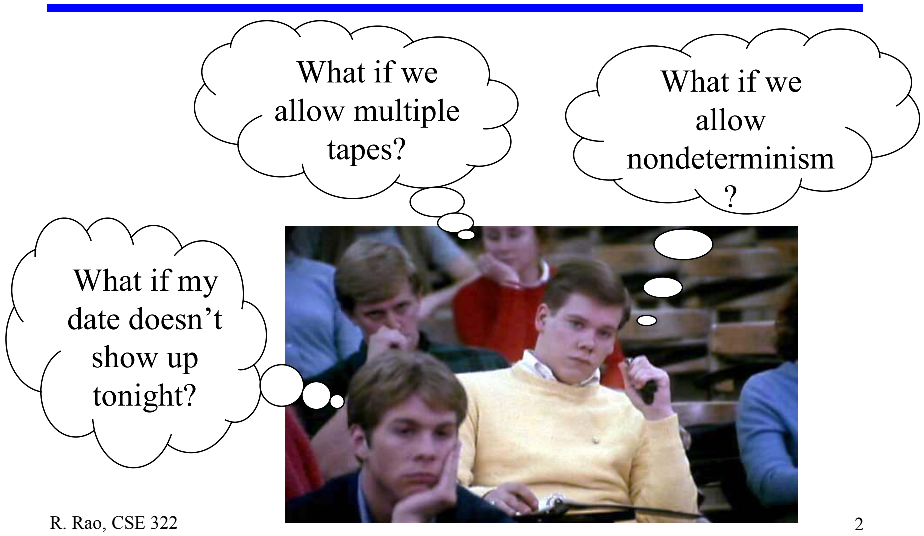
Today's Topic:
Up close and personal with Turing Machines

Can we augment the computing power of Turing machines with various accessories?

Varieties of TMs

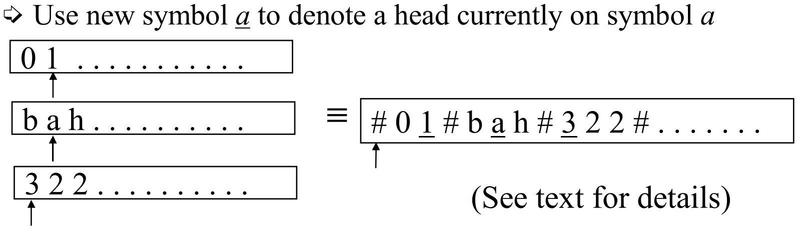


Various Types of TMs

- **Multi-Tape TMs**: TM with k tapes and k heads
  - $\delta: Q \times \Gamma^k \to Q \times \Gamma^k \times \{L,R\}^k$
  - $\delta(q_i, a_1, \ldots, a_k) = (q_j, b_1, \ldots, b_k, L, R, \ldots, L)$
- **Nondeterministic TMs (NTMs)**
  - $\delta: Q \times \Gamma \to Pow(Q \times \Gamma \times \{L,R\})$
  - $\delta(q_i, a) = \{(q_1, b, R), (q_2, c, L), \ldots, (q_m, d, R)\}$
- **Enumerator TM for L**: Prints all strings in L (in any order, possibly with repetitions) and only the strings in L
- Other types: TM with Two-way infinite tape, TM with multiple heads on a single tape, 2D infinite tape TM, Random Access Memory (RAM) TM, etc.

Surprise!
All TMs are born equal…

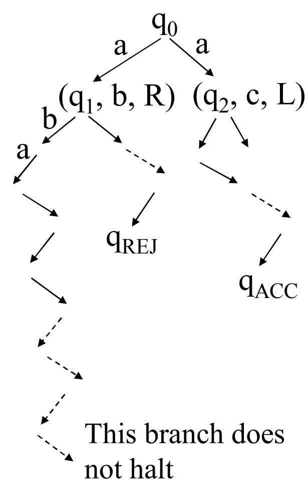
- Each of the preceding TMs is equivalent to the standard TM
  - They recognize the same set of languages (the Turing-recognizable languages)
- Proof idea: Simulate the "deviant" TM using a standard TM
- Example 1: Multi-tape TM on a standard TM
  - Represent k tapes sequentially on 1 tape using separators #
  - Use new symbol $\underline{a}$ to denote a head currently on symbol *a*



(See text for details)

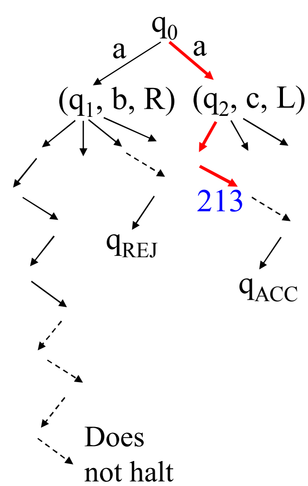
Example 2: Simulating Nondeterminism

- **Any nondeterministic TM N can be simulated by a deterministic TM M**
  - NTMs: $\delta: Q \times \Gamma \rightarrow Pow(Q \times \Gamma \times \{L,R\})$
  - No ε transitions but can simulate them by reading and writing same symbol
  - N accepts w iff there is at least 1 path in N's tree for w ending in q_{ACC}
- **General proof idea**: Simulate each branch sequentially
  - Proof idea 1: Use depth first search? No, might go deep into an infinite branch and never explore other branches!
  - Proof idea 2: Use breadth first search Explore all branches at depth *n* before *n+1*



Simulating Nondeterminism: Details, Details

- Use a 3-tape DTM M for breadth-first traversal of N's tree on w:
  - Tape 1 keeps the input string w
  - Tape 2 stores N's tape during simulation along 1 path (given by tape 3) up to a particular depth, starting with w
  - Tape 3 stores current path number E.g. ε = root node q_0 213 = path made up of 3^{rd} child of 1^{st} child of 2^{nd} child of root
- See text for more details



What about other types of computing machines?

- Enumerator TMs (or Printer Machines)
- TMs with 2-Way Infinite Tape
- TMs with Multiple Read/Write Heads
- TMs with 2-Dimensional Tape
- TMs with Random Access Memory (RAM)

The Church-Turing Thesis

- Various definitions of "algorithms" were shown to be equivalent in the 1930s
- **Church-Turing Thesis**: "The intuitive notion of algorithms equals Turing machine algorithms"
  - Turing machines serve as a precise formal model for the intuitive notion of an algorithm
- "Any computation on a digital computer is equivalent to computation in a Turing machine"

Questions to Ponder Over

Are there problems for which no algorithm exists?

Are there languages that are not even recognizable?

Does life exist elsewhere in the universe?

Next Class: Get ready fa Chapta 4 and undecidability…

Have a great nondeterministic weekend!