Decidable languages

Atri Rudra
May 26

Announcements

- Handouts
  - Sample final
  - List of topics for the finals
  - H/W #8
    - Remember your lowest H/W grade will be dropped
- Turn in your H/W #7
- Pick up graded H/W #6 at end of class

Please remember…

- I want to show you the “cool” stuff
  - There are problems that are “unsolvable”

Stop me if I am going too fast

Today’s puzzle

- Show that the following language is decidable

\[
\{ \langle G \rangle \mid G \text{ is a CFG and } 1^* \subseteq L(G) \}
\]

Last lecture

- Three things that a TM can do on an input
  - Halt and accept
  - Halt and reject
  - Loop
- A TM is a decider if it halts on all inputs

A couple of classes of languages

- \( L \) is Turing-recognizable
  - Exists a TM that accepts exactly the strings in \( L \)
- \( L \) is decidable
  - Exists a decider that accepts exactly the strings in \( L \)
A. Rudra, CSE322 7

A DFA is decidable

- \( A_{DFA} = \{ (B, w) \mid B \text{ is a DFA and } B \text{ accepts } w \} \)
- \( M = \text{"On input } (B, w) \text{"} \)
  - Check if \( B \) is indeed a DFA
  - Simulate \( B \) on \( w \)
  - If the simulation leads \( B \) to a final state then accept else reject.”

This is a decider as the simulation always terminates

A. Rudra, CSE322 8

Questions?

A. Rudra, CSE322 9

You have a choice

The choices are...

- Red pill
  - Go through some more decidable languages quickly
  - Spend most time on diagonalization
- Blue pill
  - Spend more time on some decidable languages
  - Do as much of diagonalization as possible

Will spend more time on decidable languages next week

Both topics are in the homework and no class on Monday

You have a choice

The choices are...

- Red pill
  - Go through some more decidable languages quickly
  - Spend most time on diagonalization
- Blue pill
  - Spend more time on some decidable languages
  - Do as much of diagonalization as possible

Both topics are in the homework and no class on Monday

A. Rudra, CSE322 10

If you chose the red pill

- \( E_{DFA} = \{ (A) \mid A \text{ is a DFA and } L(A) = \emptyset \} \)
- \( E_{DFA} \text{ is decidable.} \)
  - Construct a decider
  - \( T = \text{"On input } (A) \text{, where } A \text{ is a DFA} \)
    - Mark start state of \( A \)
    - Repeat until no new state gets marked
      - If \( p \) is marked and \( q \) is not, mark \( q \)
      - If no final state is marked accept else reject.”

Checking if a final state is reachable from the start state

A. Rudra, CSE322 11

Another example

- \( EQ_{DFA} = \{ (A, B) \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B) \} \)
- \( EQ_{DFA} \text{ is decidable} \)
  - \( F = \text{"On input } (A, B), \text{ where } A \text{ and } B \text{ are DFAs} \)
    - Minimize \( A \) and \( B \) to \( A' \) and \( B' \)
    - Check if \( A' \) and \( B' \) are isomorphic
      - If they are then accept
      - Else reject.”

Alternate proof in Sipser: uses \( E_{DFA} \) as a sub-routine