CS581 Theory of Computation: Chapter 4 review

Due on February 22 2016 at 2:00pm

Harry H. Porter Winter 2016

Konstantin Macarenco

Decidable languages

1. Building deciders for various languages
2. $A_{DFA} = \{\langle B, w\rangle | B$ is a DFA that accepts $w\}$ - Is decidable.
   On input "w" construct B, accept if B accepts, reject if B rejects.
3. $A_{NFA} = \{\langle N, w\rangle | N$ is a NFA that accepts $w\}$ - Is decidable.
   On input "w" convert N into DFA B, run A_{DFA} on $\langle B, w\rangle$ accept if A_{DFA} accepts, otherwise reject.
4. $A_{REG} = \{\langle R, w\rangle | R$ is a Regex that generate $w\}$ - Is decidable.
   On input "w" convert R into NFA N, run A_{NFA} on $\langle N, w\rangle$ accept if A_{NFA} accepts, otherwise reject.
5. $E_{DFA}\{\langle B\rangle | B$ DFA and L(B) $= \emptyset\}$
   On input $\langle B\rangle$ Recursively mark all the states reachable from the start state. If no accept state is marked - accept, otherwise reject.
6. $EQ_{DFA}\{\langle A, B\rangle | A, B$ are DFAs and L(A) = L(B)$\}$
   On input $\langle A, B\rangle$ construct symmetric difference C of A and B, and run E_{DFA} on C, accept if E_{DFA} accepts, and reject otherwise.
7. $A_{CFG} = \{\langle C, w\rangle | C$ is a CFG that generates $w\}$ - Is decidable.
   On input "w" convert C into chomsky normal form C', generate all strings that can be generated in $2n-1$ steps, where n is the lenght of w, if $w \in Generated$, accept, otherwise reject.
8. $E_{CFG}\{\langle C\rangle | C$ CFG and L(C) $= \emptyset\}$ - is decidable
   On input $\langle C\rangle$. Recursively mark all the strings that can lead to terminals. If start string is marked reject, otherwise accept.
9. $EQ_{CFG}\{\langle A, B\rangle | A, B$ are CFGs and L(A) = L(B)$\}$ - is not decidable

10. Every Question about Regular languages is decidable.
11. Some questions about CFLs are decidable, but some are not.
12. The set of all Turing machines is countably infinite.
13. The set of all Turing-Recognizable languages if countably infinite.
14. The set of all Languages is uncountably infinite.

Undecidable languages

1. Undecidable examples
   1. Is a CFG ambiguous
   2. Do two CFGs have any strings the can generate in common?
   3. Is the complements of a cfg also a cfg?
2. $A_{TM} = \{\langle M, w\rangle | M$ is a TM that accepts $w\}$ - Is turing recognizable. (Universal Turing machine)
   On input "w" construct and run M, accept if M accepts, reject or halt if M rejects or halts.
3. Countability proof by Cantors diagonalization method.
4. Onto (every element in A has corresponding element in B) vs one-to-one(every element in A corresponds to a unique element in B) function.
5. Correspondence - onto and one-to-one.
6. Infinite number of languages is infinite and not countable, number of TM is infinitely countable therefore some languages are not Turing Recongnizable
7. Halting problem - proof that A_{TM} is not decidable.

a. Assume that A_{TM} is decidable.
b. Construct TM H that decides A_{TM}
   H on input $\langle M, w\rangle$ accepts if M accepts, and rejects if M rejects.
c. Construct a TM D that takes a description of a turing machine as an input $\langle M\rangle$, and uses H as subroutine in the following way:
   It runs H on inputs $\langle M, M\rangle$, and accepts if H rejects, and rejects if H accepts.
d. Run D on description of itself the we get a contradiction:
   D accepts, when D rejects D, and
   D rejects, when D accepts D.
   PARADOX
e. Hence we get a contradiction, therefore A_{TM} is not Decidable.

8. If complement of a language is recognizable then Language is called Turing Co-recognizable.
9. If a languages is turing recognizable and turing co-recognizable, then this language is decidable.