COMP 598

Talise Wang 260829722

May 28, 2020

Question 1

It is decidable for a given input.
Define the Turing machine M as $M = (Q, \Sigma, \Gamma, \delta, q_0)$ a deterministic turing machine. Since it never use more than 598 cells, there are finite configurations in that machine. Let the number of these configurations be n. Then we will get $n = \underbrace{598}_{\text{where the head can be}} \cdot \overbrace{|Q|}^{\text{states}} \cdot \underbrace{|\Gamma|^{598}}_{\text{content on the tape}}$
So there will be only 2 situations.
The first situation is: the machine stays in the 598 cells, which means there are some configurations repeated in the first n steps by the pigeonhole thm. Since we define M as a deterministic turing machine. So it will loop when it meets the repeated configuration. Which means the machines will always stay in the 598 cells.
The second situation is: the machine will stay out of the 598 cells, which simply means that there will be no repeated configuration. Otherwise, it will be the first situation. And it must be done in $n + 1$ steps.
So we proved!

Question 2

We know that the algorithm to check that if a word is in a CFL is decidable and will finally halt.

so we just have to run this algorithm in $L(G_1) \cap L(G_2)$ on every word. We can get a "No" answer if we can find a common word. But, in this case, the "Yes" answer may or may not happen.
To show it is not decidable. First, we know that $VALCOMPS2(M, w)$ is empty iff M does not accept w. So we can do this by reducing it to $\neg A_{TM}$, which is undecidable(proved in the lecture notes).
So we proved!

Question 3

To get the position of the submarine. We first define that it started at (x, y). Define the speed is v. For the direction, let's guess it is up, so we can get the position of the submarine at the n steps as $(x, y + n * v)$. Similarly, for the dirction "down", we will get $(x, y - n * v)$ at the n steps. For "right", it's $(x + n * v, y)$ at n steps and for "left", it's $(x - n * v, y)$ at n stpes.
So we can try all the combinations above, we will finally hit the submarine.
To achieve this, we will use the dovetailing. Define a function: $f : Z \times Z \to N$, we use this to map x & y to a natural number a. Similarly, map a and v(the speed) to a natural number b, and map b with the dirction character,
Now, let t be the time, at every t,do the following: Use $\lfloor t/4 \rfloor$ as the n and decode x, y, v Use $t \bmod 4$ as the direction (define 0 for up, 1 for left, 2 for right, 3 for down). Then Zap!
So we can programmatically try out all possible guesses which guarantee a hit.

Question 4

4.1

Define M as $M = (Q, \Sigma, q_0, \delta, F)$ a DFA that recognizes L.
Then define a new DFA $M' = (Q, \Sigma, q_0, \delta, F')$ where $F' = \{s \in Q | \delta^*(s, w) \in F\}$. Q, Σ, q_0, δ are the same as M. Clearly, it recognizes L/w, so L/w is regular.

4.2

We know that N is context-free, so $N\#\Sigma^*$ and $\Sigma^*\#L(G)$ are also context free. Since the union of two context free languages are context free. So L is context free!
To prove L is regular iff $L(G) = \Sigma^*$, we now have two cases:
If $L(G) = \Sigma^*$, then $L = \Sigma^*\#\Sigma^*$. Clearly, L is regular.
If $L(G) \neq \Sigma^*$, then we can find w that $w \in \Sigma^*$ while $w \notin L(G)$. Let's consider $L/\#w$, it's just N. If N is not regular, so $L/\#w$ is not regular so L is not regular.
So we proved! L is regular iff $L(G) = \Sigma^*$.
But $L(G) = \Sigma^*$ is undecidable, so L is regular is also undecidable.

Question 5

5.1

This one is undecidable.
R is a regular language. If we can decide whether $R \subseteq L$, then it is the same as we can decide whether a arbitrary regular language $\subseteq L$. Since Σ^* is regular, this also equivelent to whether we can decide $L = \Sigma^*$. But clearly, we can not decide that since L is a context free language. So it's undecidable.

5.2

This one is decidable.
If $L \subseteq R$, then $L \cap \overline{R} = nil$ must hold. Since R is a regular language, then $\overline{R}$ is also a regular language. With L is a context free language, we can know that $L \cap \overline{R}$ is also context free. Since whether it is nil or not is decidable, So we proved!

Question 6

I choose python:

```
print(open(__file__).read())
```