Erli Cai

Question 1: Out of Context

a. $S \rightarrow \begin{bmatrix} b \\ b \\ q_s \end{bmatrix} \begin{bmatrix} \$ \\ \$ \\ \ \end{bmatrix}, \quad \begin{bmatrix} b \\ b \\ q_s \end{bmatrix} \rightarrow \begin{bmatrix} b \\ b \\ q_s \end{bmatrix} \begin{bmatrix} a \\ a \\ \ \end{bmatrix}$ for all $a, b \in \Sigma$

b. Now we try to simulate M (based on $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L.R\}$)
if we have $\delta(q, A) = (p, B, L)$,then we add in our production rule:
$\begin{bmatrix} c \\ C \\ \ \end{bmatrix} \begin{bmatrix} a \\ A \\ q \end{bmatrix} \rightarrow \begin{bmatrix} c \\ C \\ p \end{bmatrix} \begin{bmatrix} a \\ B \\ \ \end{bmatrix}$ for all $c, C \in \Gamma$
if we have $\delta(q, A) = (p, B, L)$,then we add in our production rule:
$\begin{bmatrix} a \\ A \\ q \end{bmatrix} \begin{bmatrix} c \\ C \\ \ \end{bmatrix} \rightarrow \begin{bmatrix} a \\ B \\ \ \end{bmatrix} \begin{bmatrix} c \\ C \\ p \end{bmatrix}$ for all $c, C \in \Gamma$

c. $\begin{bmatrix} n \\ N \\ \ \end{bmatrix} ...... \begin{bmatrix} a \\ A \\ \ \end{bmatrix} \begin{bmatrix} b \\ B \\ q_{accept} \end{bmatrix} \begin{bmatrix} c \\ C \\ \ \end{bmatrix} ...... \begin{bmatrix} z \\ Z \\ \ \end{bmatrix} \rightarrow n....abc....z$ This is the only production rule that doesn't generate non-terminate, so grammar produce a string x if and only if M reaches accepted state

Question 2: Tough Decisions

a.Given DFA D, $\{< D > |$ L(D) is finite$\}$ is decidable.
Suppose D has n states, TM search for all the path of n+1 steps. If any state can be visited repeatedly is one path, then mark that state. If in any path, any accepted state is reached passing through a marked state, then we reject. Else, accept.

b. This is recognisable but not decidable. Recognisable because, we can simulate M on all possible strings and accept if any halt.
It is not decidable. Because if it is decidable we know when TM never halts and then $\{< M, x > | MisTuringmachine, andMacceptx\}$ would be decidable (which we have shown in class is not decidable).

c. This is not recognisable. Since M could loop on x, we never know if it will read all of x, so not recognisable.

Question 3: Closure Properties

a. This won't work. Suppose M_A loops on x, while M_B accepts x. since M_A loops on x, we never get to step 2. So the TM never halts while it should accept x.
We could build the Turing machine do the following, Alternate simulate M_A, M_B on x, if one of them accept, then accept.

b. This will work. Since both machines (T_A, T_B) need to halt and accept x for x to be accepted by our new machine. If either machine loops or rejects, the new machine won't accept.