1. Give a context free grammar for the following language:

$$L = \{a^n b^m c^k \mid k \neq n + m\}$$

[5]

$S \to X_1 \mid X_2$

X_1: $n+m < k$

X_2: $n+m > k$

$X_1 \to C_1 \mid aX_1C_1 \mid bB_1C_1$

$B_1 \to bB_1C_1 \mid C_1$

$C_1 \to c \mid C_1c$

$X_2 \to aX_2C_2 \mid bB_2C_2 \mid B_2 \mid a$

$B_2 \to bB_2C_2 \mid b$

$C_2 \to c \mid \epsilon$

Another solution:

$S \to S_1 \mid A_2$

$n+m > k$ $\quad S_1 \to aA_1 \mid bB_1$

$A_1 \to aA_1C_1 \mid B_1 \mid aC_1 \mid \epsilon$

$B_1 \to bB_1C_1 \mid bC_1 \mid \epsilon$

$C_1 \to c \mid \epsilon$

$n+m < k$ $\quad A_2 \to aA_2c \mid B_2$

$B_2 \to bB_2c \mid C_2$

$C_2 \to C_2c \mid c$

2. Let

$$L_1 = \{a^n b^m c^k \mid n, m, k \geq 0\}$$

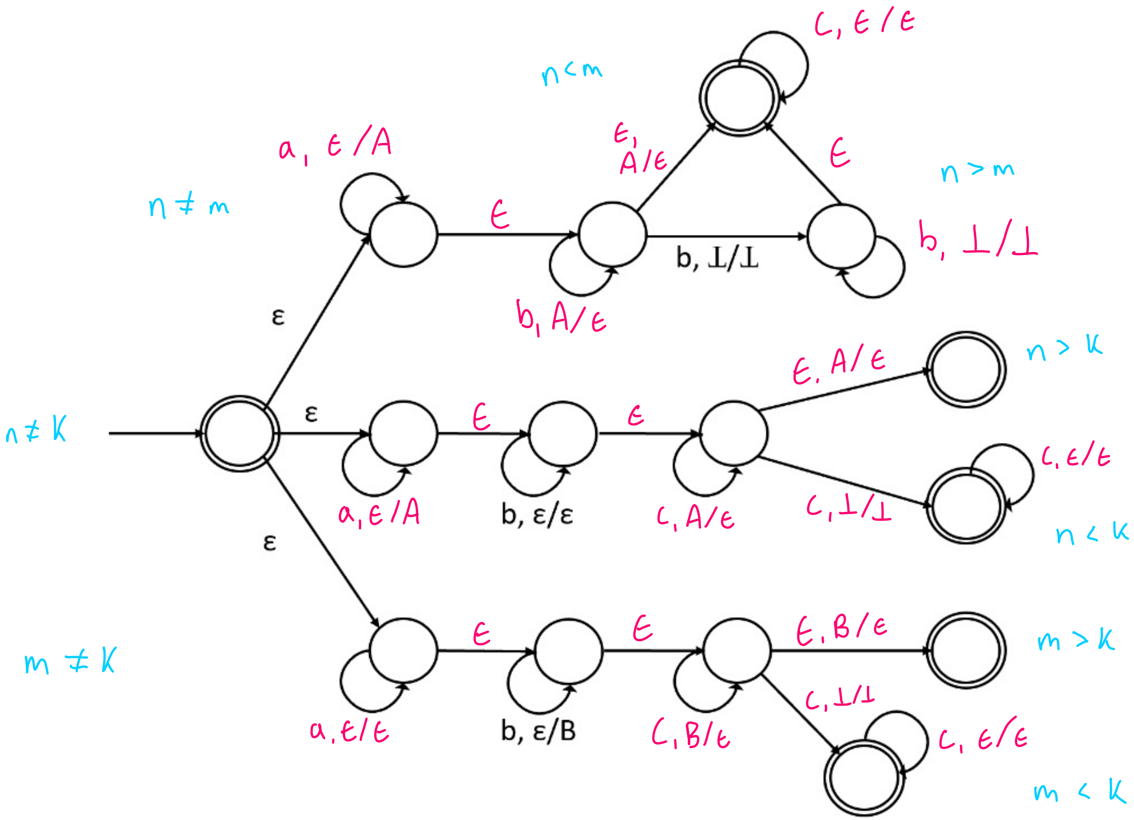
and let

$$L_2 = \{a^n b^n c^n \mid n \geq 1\}$$

Complete the pushdown automata M (in the figure below), such that

$$L(M) = L_1 - L_2,$$

where $\Sigma = \{a, b, c\}$. Your solution must use the template below. You may **not** add/remove transitions. If add/remove states/transitions, or change the labelled transitions already there, you will receive 0 on the question. [10]



3. Turing machines can also be used to output more than just Accept/Reject. When they halt, they may also have useful output written on the tape. For example, a Turning machine may

add numbers by taking in strings such as: 101#111 and having the string 1100 written on the tape when is halts. Note, 1100 equals 101 + 111. You are going to create such a machine. To do this your machine will have three distinct parts. The first part will take the input of the form $B_1\#B_2\#$, where B_1 and B_2 are binary sequences of equal length, and complete its portion with $X\#X\#C$ written on the tape and the read/write head pointing to the left most digit. The sequences C and X are the same length as B_1 and B_2. Moreover, the ith character of C will be a c if the ith digits of B_1 and B_2 are both 1's, will be a 0 the ith digits of B_1 and B_2 are both 0's, and be 1 otherwise. The sequence X is just a sequence of x's. To help you understand what this portion is doing, the c stands for *carry-bit*. As an example, if the input is

10100#10101#

then this part of the machine will complete its task with the tape looking like:

$xxxxx\#xxxxx\#c0c01$

For this portion of the machine reverse engineer my partial solution below and fill-out all *TODO* values in the table.

	1	0	x	#	c	☐
q_s	($q_{1,1}$, x, R)	($q_{1,3}$, x, R)	(q_s, x, R)	-	-	-
$q_{1,1}$	($q_{1,1}$,1,R)	($q_{1,1}$,0,R)	-	($q_{1,2}$, #, R)	-	-
$q_{1,2}$	($q_{1,5}$,x,R)	($q_{1,6}$, x, R)	($q_{1,2}$, x, R)	-	-	-
$q_{1,3}$	($q_{1,3}$,1,R)	($q_{1,3}$,0,R)	-	($q_{1,4}$, #, R)	-	-
$q_{1,4}$	($q_{1,6}$, x, R)	($q_{1,7}$,x,R)	($q_{1,4}$, x, R)	-	-	-
$q_{1,5}$	($q_{1,5}$,1,R)	($q_{1,5}$,0,R)	-	($q_{1,8}$,#,R)	-	-
$q_{1,6}$	($q_{1,6}$,1,R)	($q_{1,6}$,0,R)	-	($q_{1,9}$,#,R)	-	-
$q_{1,7}$	($q_{1,7}$,1,R)	($q_{1,7}$,0,R)	-	($q_{1,10}$,#,R)	-	-
$q_{1,8}$	($q_{1,8}$, 1, R)	($q_{1,8}$, 0, R)	-	-	($q_{1,8}$, c, R)	($q_{1,end1}$, c, L)
$q_{1,9}$	($q_{1,9}$, 1, R)	($q_{1,9}$, 0, R)	-	-	($q_{1,9}$, c, R)	($q_{1,end1}$, 1, L)
$q_{1,10}$	($q_{1,10}$, 1, R)	($q_{1,10}$, 0, R)	-	-	($q_{1,10}$, c, R)	($q_{1,end1}$, 0, L)
$q_{1,end1}$	($q_{1,end1}$,1,L)	($q_{1,end1}$,0,L)	-	($q_{1,end2}$,#,L)	($q_{1,end1}$,c,L)	-
$q_{1,end2}$	($q_{1,end3}$,1,L)	($q_{1,end3}$,0,L)	($q_{1,end2}$,x,L)	($q_{1,end2}$,#,L)	-	($q_{2,s}$,☐,R)
$q_{1,end3}$	($q_{1,end3}$,1,L)	($q_{1,end3}$,0,L)	($q_{1,end3}$,x,L)	($q_{1,end3}$,#,L)	-	(q_s,☐,R)

For the next part of the machine, we want to clean up the tape a bit. That is, we want to go from something below, where the read/write head is at the far left end

$$xxxxxx\#xxxxxx\#c0c01$$

to

$$c0c01$$

where the read write head is at the far right end. Complete the table below to achieve this.

The final and third part of the machine has a string like

$$c0c01$$

	1	0	x	#	c	□
$q_{2,s}$	-	-	$(q_{2,s}, \square, R)$	$(q_{2,1}, \square, R)$	-	-
$q_{2,1}$	-	-	$(q_{2,1}, \square, R)$	$(q_{2,2}, \square, R)$	-	-
$q_{2,2}$	$(q_{2,2}, 1, R)$	$(q_{2,2}, 0, R)$	-	-	$(q_{2,2}, c, R)$	$(q_{3,s}, \square, L)$

written on the tape with the read/write head on the far right side. After a single pass the machine will have the final binary string (the addition of the two input written on the tape). Following our example when the machine halts the tape will look like

$$101001$$

Produce and complete a transition table for this portion. You should use the fewest number of states possible.

[10] marks for the first table, [2] marks for the second table, and [3] marks for the final table.

	1	0	c	□
$q_{3,s}$	$(q_{3,s}, 1, L)$	$(q_{3,s}, 0, L)$	$(q_{3,1}, 0, L)$	End
$q_{3,1}$	$(q_{3,1}, 0, L)$	$(q_{3,s}, 1, L)$	$(q_{3,1}, 1, L)$	$(q_{3,s}, 1, L)$

no carry over

carry over