Q1) i) If A is a regular language then there exists a number P (pumping length) where if S is a string from A with length at least P ($|S| \geq P$) then S can be divided into three pieces, $S = xyz$ satisfying the following conditions:

(i) $\forall i \geq 0$ the value $xy^iz \in A$

(ii) $|y| > 0$ $(y \neq \epsilon)$

(iii) $|xy| \leq P$

(ii) $L = \{a^n \mid n$ is a prime $\}$

Let L is regular

Let P be the pumping length.

$L = \{aa, aaa, aaaaa, \dots\}$

Let $P = 3$

$S = aaa$

$|S| = 3$

$S \geq P$ (True) Let $x = a$ $y = a$ $z = a$

(i) $|y| > 0$

(ii) $|xy| \leq P$

$\Rightarrow 2 \leq 3$

(iii) $i \geq 0$

$xy^iz \in L$

For $i = 0$

$xy^0z = aa^0a$

$= aa \in L$

$i = 1$

$xy^1z = aa^1a$

$= aaa \in L$

$i = 2$

$xy^2z = aa^2a$

$= aaaa \notin L$

Hence, L is not a context-free language.

Q.2) i) $L = \{a^i b^j c^k \mid i, j, k \geq 0$ and $i = j$ or $i = k\}$

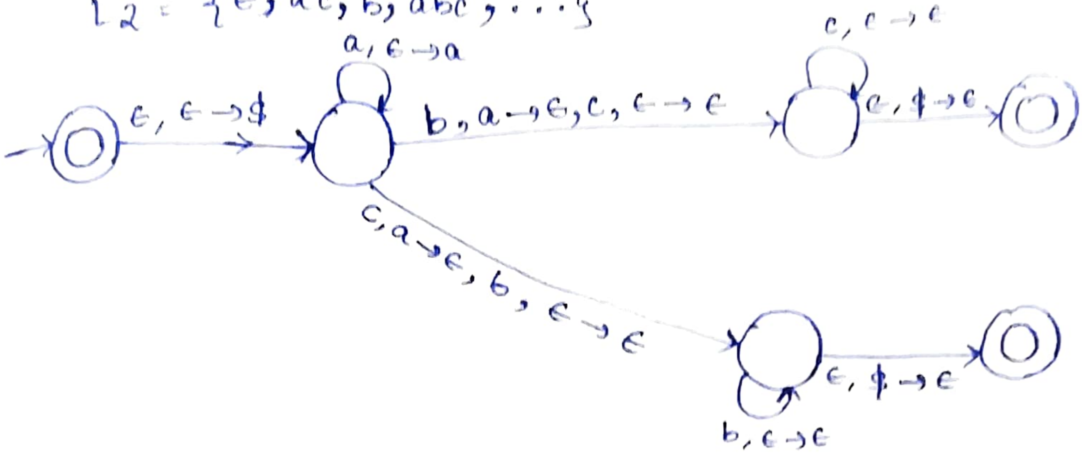
$L = L_1 \cup L_2$

$L_1 = \{a^i b^j c^k \mid i = j\}$

$L_2 = \{a^i b^j c^k \mid i = k\}$

$L_1 = \{\epsilon, c, ab, abc, \ldots\}$

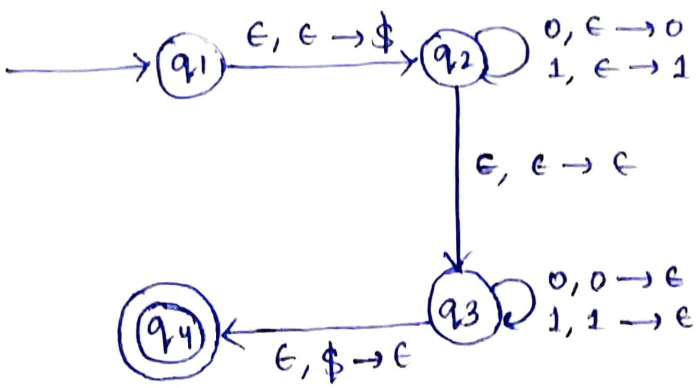
$L_2 = \{\epsilon, ac, b, abc, \ldots\}$



ii) $L = \{ww^R \mid w \in \{0,1\}^*\}$

~~L = {aa, abba, abbba, aaaa, ...~~

$L = \{00, 0110, 011110, 0000, 1111, \ldots\}$

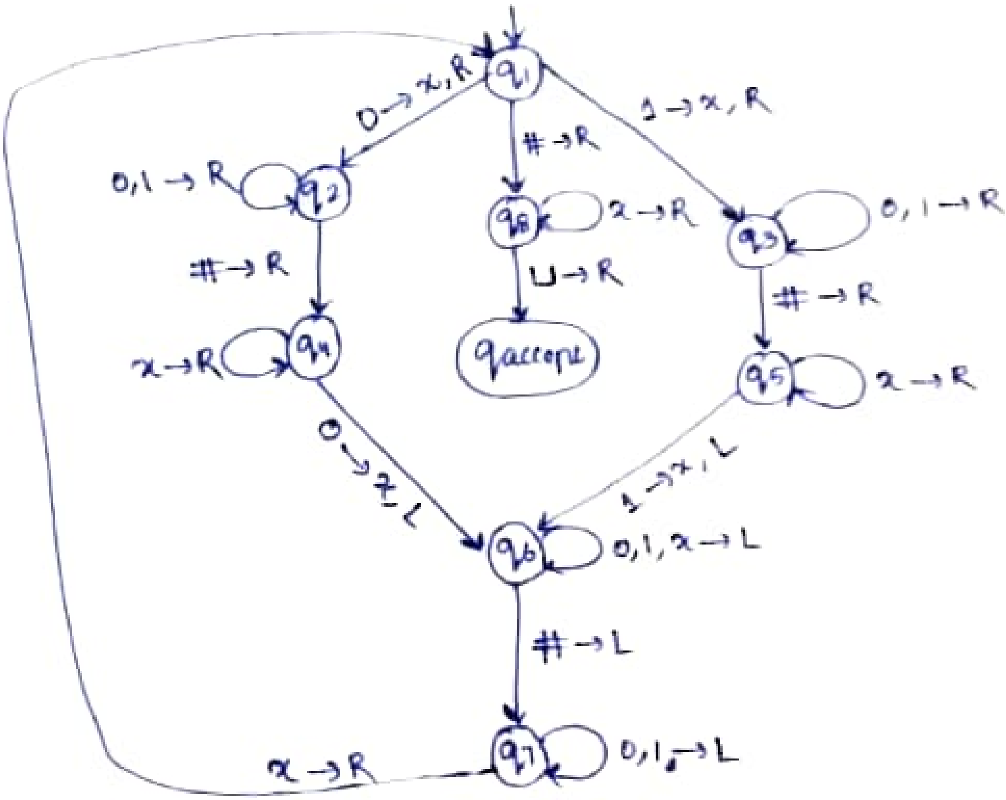


(ii) $L = \{ w\#w \mid w \in \{0,1\}^* \}$

$Q = \{ q_1, \ldots, q_8, q_{accept}, q_{reject} \}$

$\Sigma = \{0, 1, \#\}$

$\Gamma = \{0, 1, \#, x, \sqcup\}$



Q4) i) A decision problem 'p' is said to be decideable if their exist a halting turing machine that can decide or solve the problem.

eg: Decision problems wrt to FA which are decidable

→ Finiteness Problem

→ Emptyness problem – 'ϕ' → empty language.

→ Membership problem

→ Equivalance problem

→ Completeness problem : Σ^* all the 5 state ~~of~~ and final state

Undecidable: A problem is said to be undecidable if there exists no decision algo or Turing Machine that can solve the problem.

ex: Equivalence of 2 CFG.

Ambiguity of CFG.

Halting problem of Turing machine.

11) Let's assume for the purpose of obtaining a contradiction that TM R decides $HALT_{TM}$. We construct TM S to decide A_{TM}, with S operating as follows.

S = "On input<M, w>, an encoding of a TM M and a string w:

1. Run TM R on input <M, w>
2. If R rejects, reject.
3. If R accepts, simulate M on w until it halts.
4. If M has accepted, accept; if M has rejected, reject."

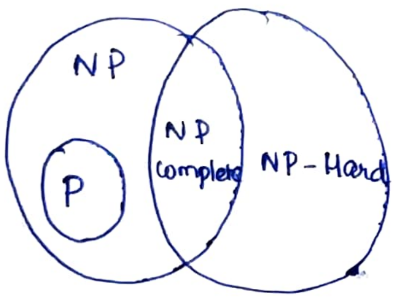
clearly, if R decides $HALT_{TM}$, then S decides A_{TM}. Because A_{TM} is undecidable, $HALT_{TM}$ also must be undecidable.

Q5) i) P: A language L is said to be in P class if there ~~this~~ exist a Deterministic Turing machine M that solves L in polynomial time.

NP: A language L is said to be NP class if there exist a Non-Deterministic Turing Machine M that can solve L in Non-Deterministic polynomial time

NP-complete: A problem L is said to be NP complete if $L \in NP$ and every problem L that belongs L is polynomial time reducable to L.

NP-hard: A problem L is said to be NP-hard if every problem L in NP is polynomial time reducable to L and L is not necessarily in NP class



ii) Reducibility: Reduction is a way of converting one problem to another problem in such a way that the sol^n to the second problem can be used to solve the first problem. So reduction always involve two problem i.e. A and B and is derived as

A α B