Faith can move the mountains

Module 5 Turing Machine and Recursive Function Theory

1 Turing Machine

(i) Turing Machine accepts / enumerates recursively enumerable languages

(ii) Turing can be described using a tuple having 7 elements viz

- (i) Q : Set of States
- (ii) Σ : Input Alphabet
- (iii) Γ : Tape Alphabet
- (iv) δ : Transition Function
- (v) q_0 : Start state
- (vi) B : Blank Symbol
- (vii) F : Final states set

(iii) Turing Machine can be deterministic or non-deterministic
Transition function for DTM is as :-

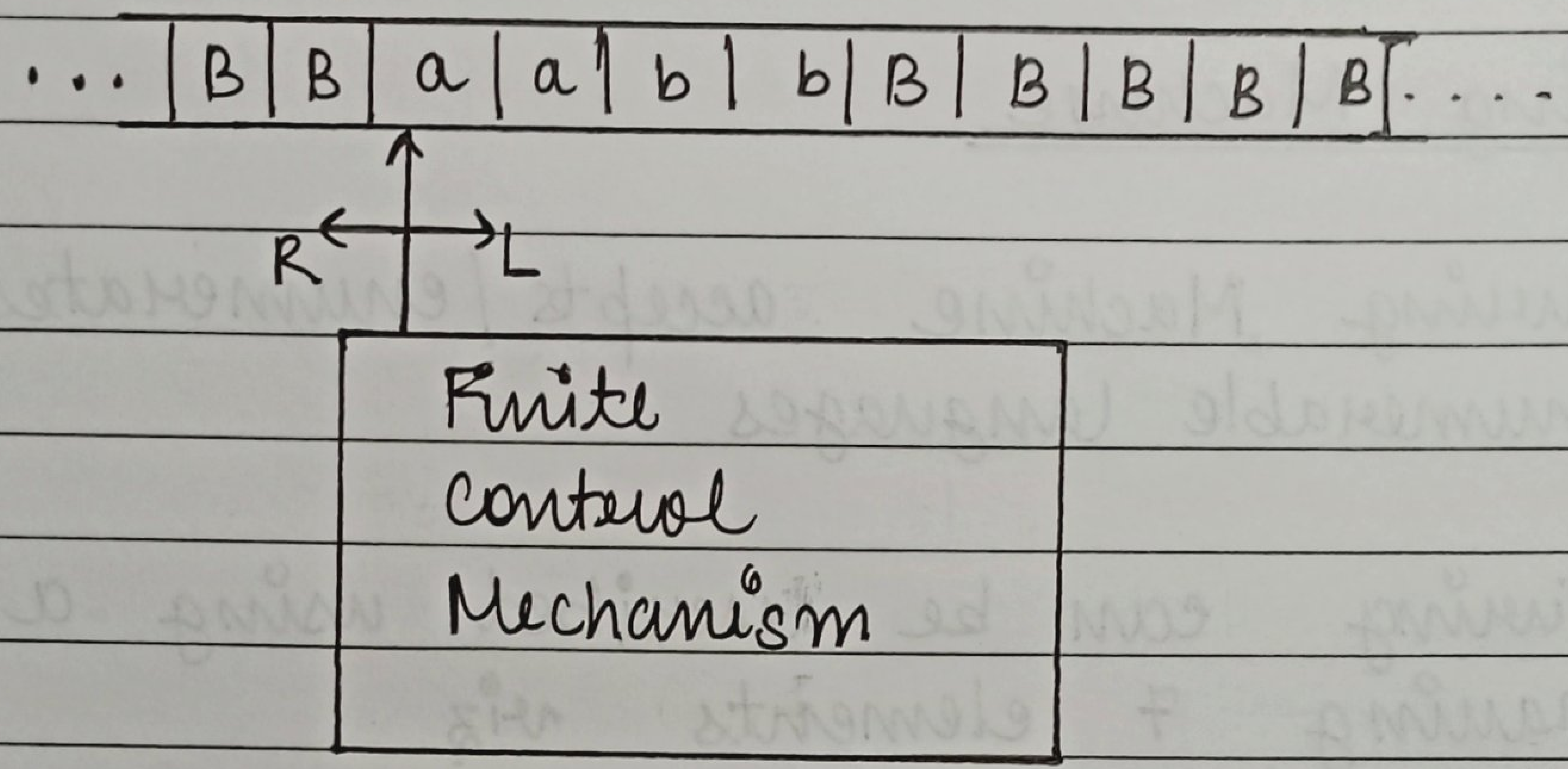
$$\delta_{DTM} : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R\}$$

where as,
Transition function for NTM is as :-

$$\delta_{NTM} : Q \times \Gamma \longrightarrow 2^{Q \times \Gamma \times \{L, R\}}$$

(iv) Turing Machine reads symbol from a tape. which as input symbol and blanks (B). The read head of turing machines can move Left (L) or Right (R) as instructed by the finite control.

(v) The general picture of TM can be shown as



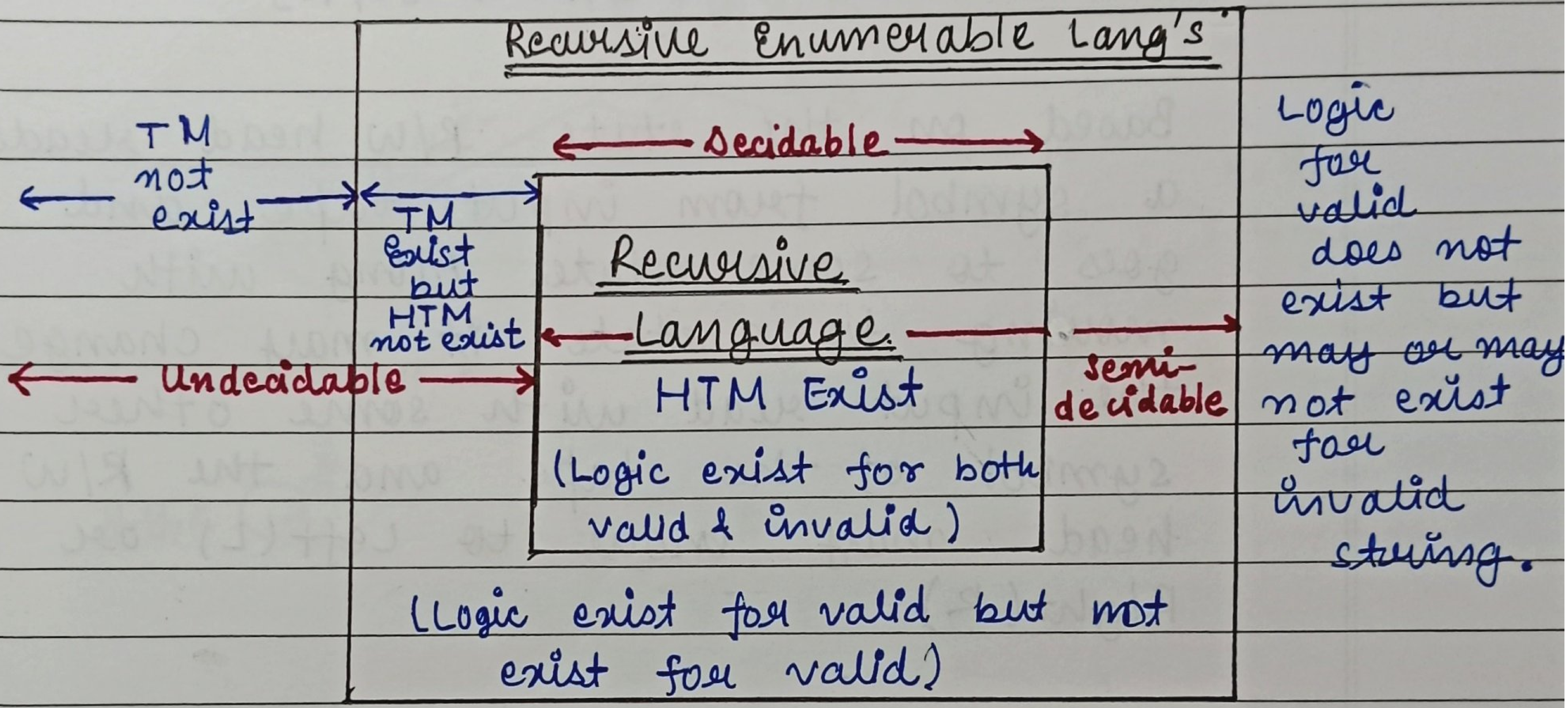
(vi) Turing Machine has the capability to read inputs from both sides as well as to write over input tape

(vii) Turing Machines accepts strings using two mechaism :-
(i) Final state Mechanism
(ii) Halting configuration

(viii) Turing machine may or may not halt for recursively enumerable languages because in REL's logic exist for valid strings and logic may or may not exist for invalid strings.

(ix) A turning machine where logic exist for both valid and invalid string will always halt and is called halting turing machine.

2 Recursively Enumerable Languages



(i) All recursively enumerable language have a Turing Machine which may or may not halt for invalid string but will halt for valid strings

(ii) Set of all recursive's are actually those REL's where logic exist for valid and invalid string

(iii) Languages where HTM exist are decidable and languages for whoom turing machine does not exist are undecidable and languages for whoom turing machine exist are semidecidable

3 Understanding Transition

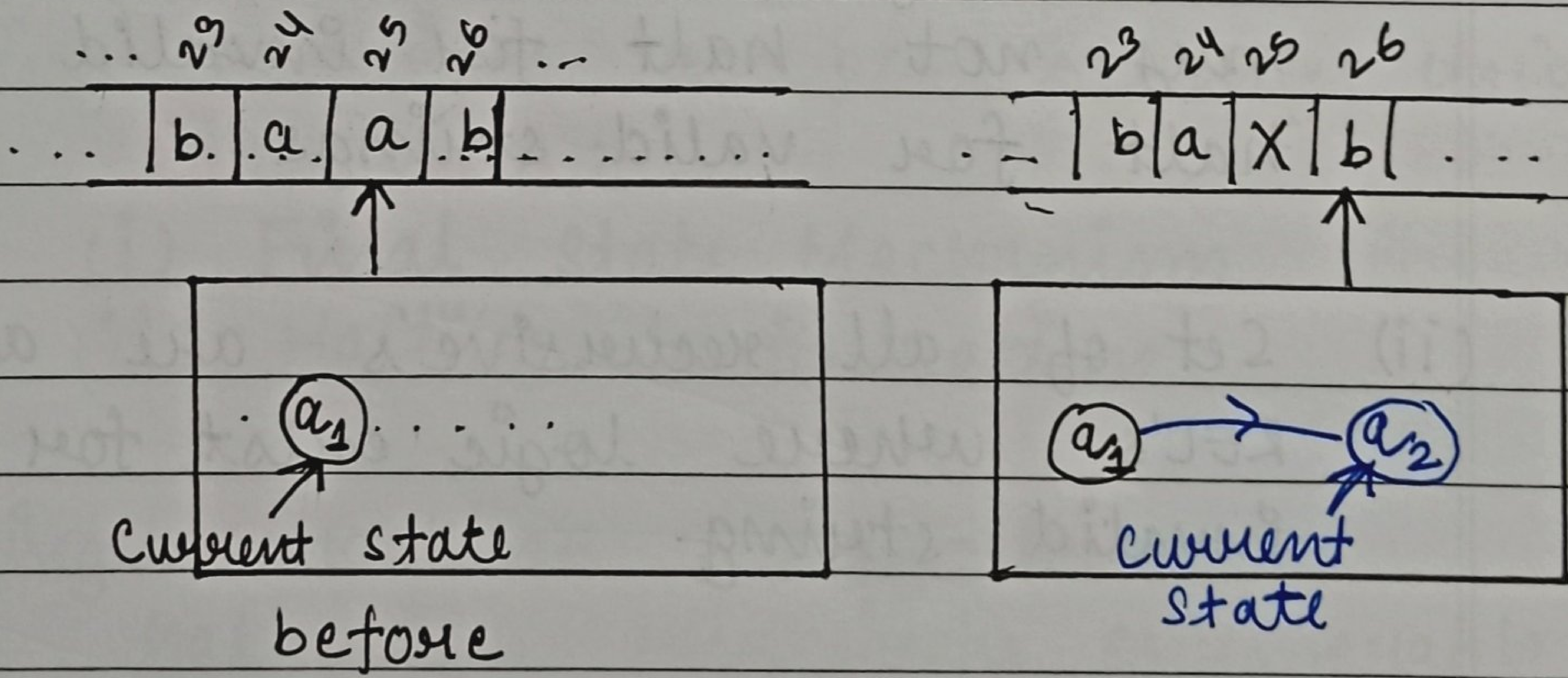
3.1 Deterministic Transitions

$$\delta : Q \times T \longrightarrow Q \times T \times \{L, R\}$$

Based on the state R/W head reads a symbol from input tape and goes to some state. Along with moving to a state it may change the input read with some other symbol of the tape and the R/W head may move to Left (L) or Right (R)

Example

$$\delta(q_1, a) := (q_2, X, L)$$



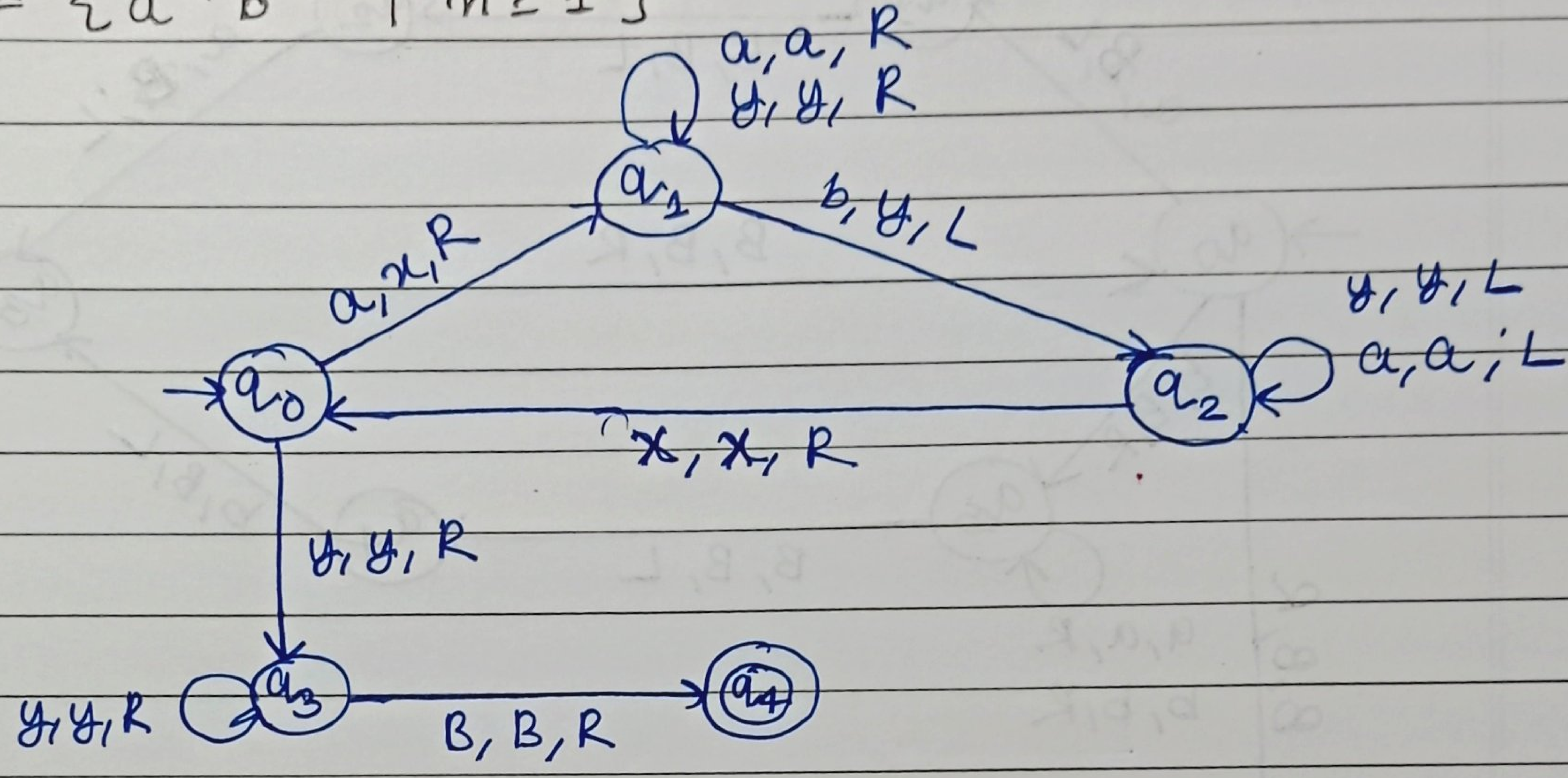
3.2 Non-Deterministic Transition

$$\delta : Q \times T \longrightarrow 2^{Q \times T \times \{L, R\}}$$

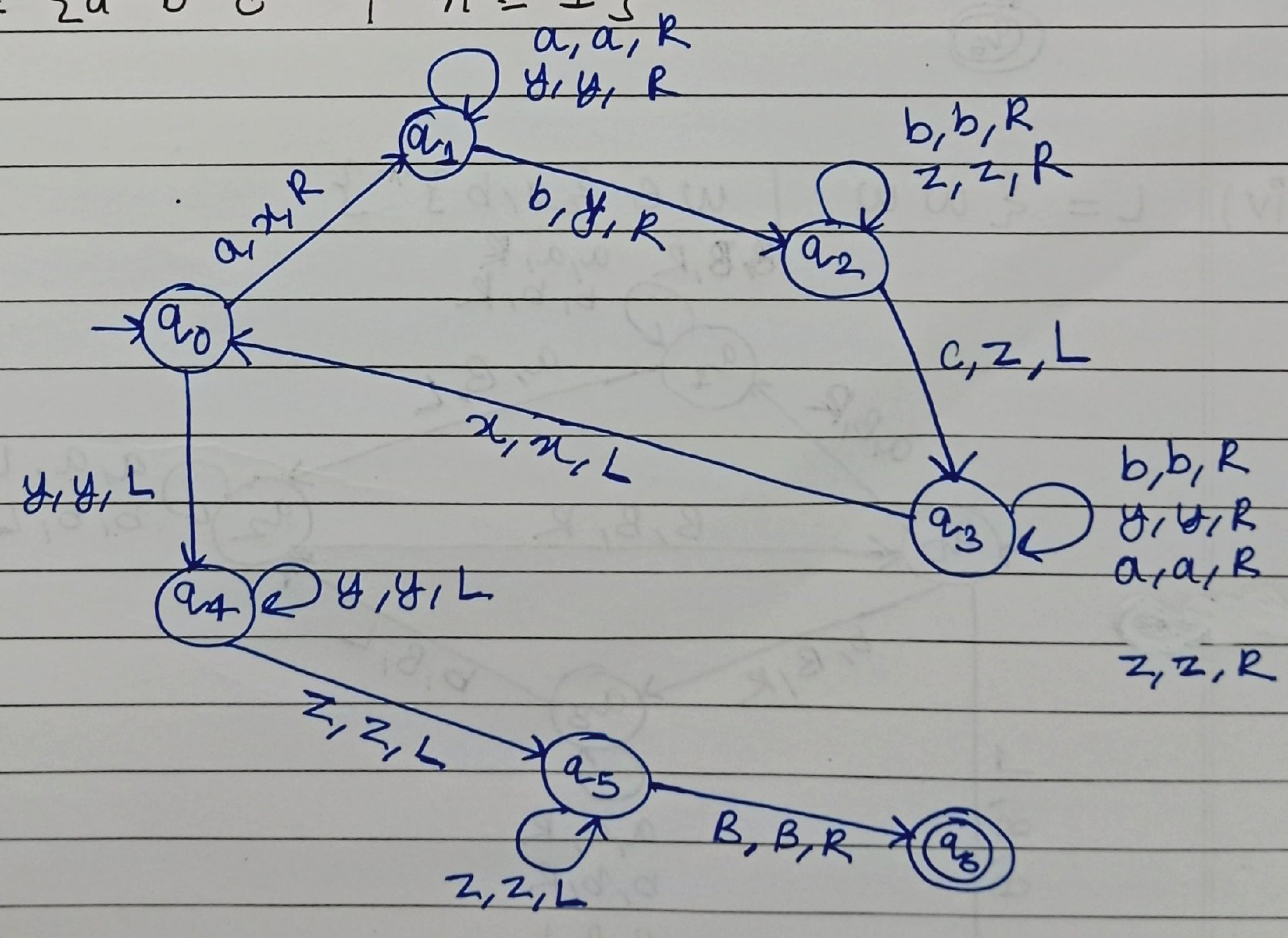
Non-deterministic Transition may not give any transition for some (Q, T) or may give 1 or more than 1 transition.

4 Construction of Turing Machine

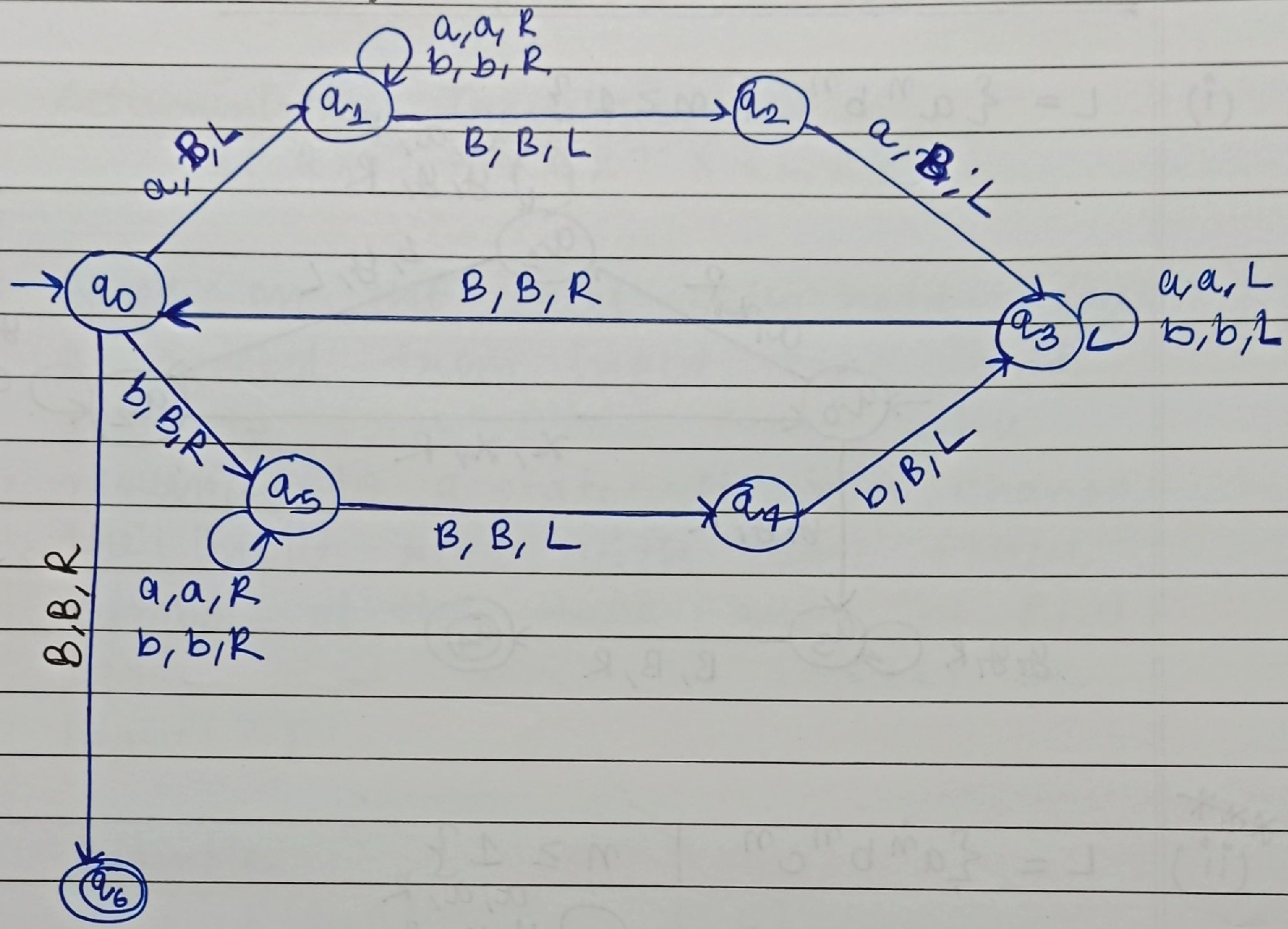
(i) $L = \{a^n b^n \mid n \geq 1\}$



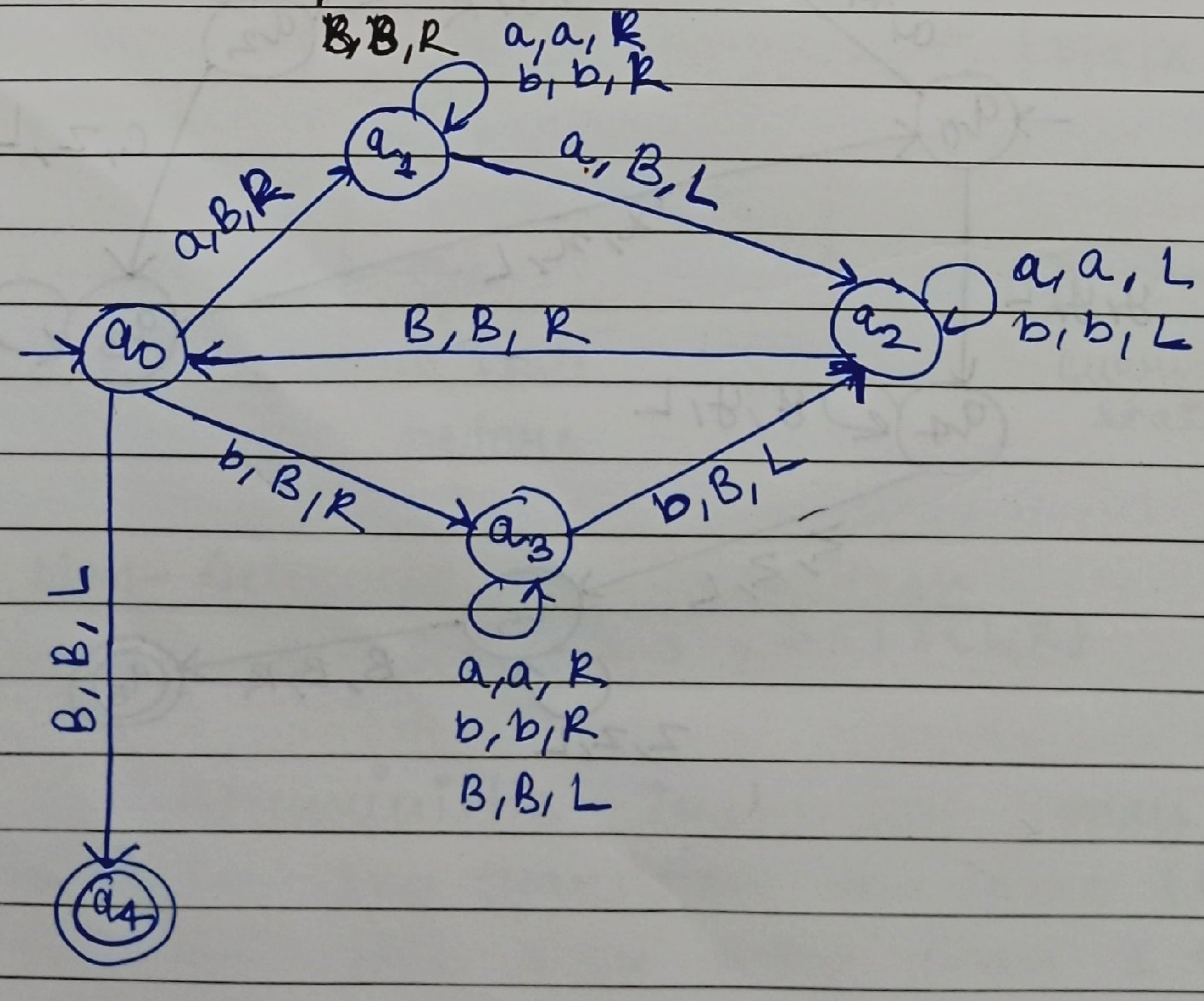
*** (ii) $L = \{a^n b^n c^n \mid n \geq 1\}$



(iii) $L = \{ w w^R \mid w \in \{a,b\}^* \}$



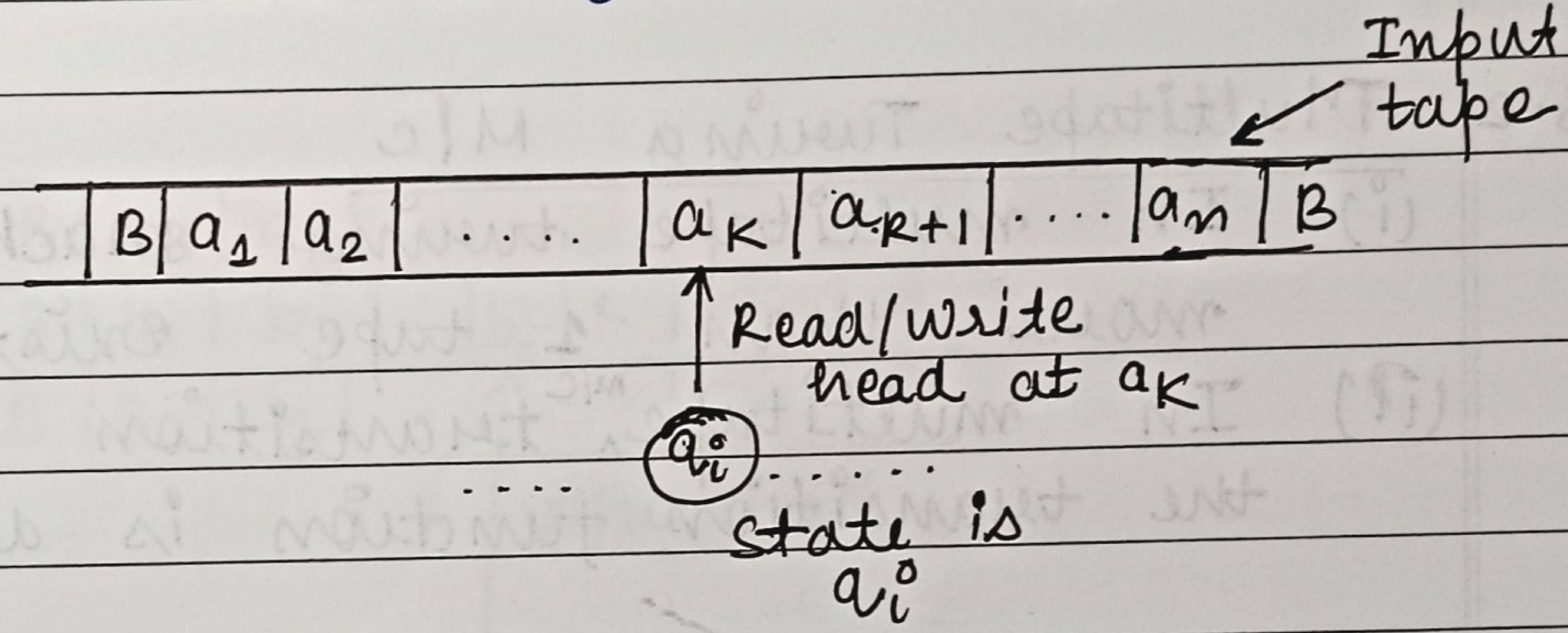
(iv) $L = \{ w w \mid w \in \{a,b\}^* \}$



5 Instantaneous Description of a TM

(i) Instantaneous description tells the current state of the TM T defined as $(Q, \Sigma, \Gamma, \delta, q_0, B, F)$

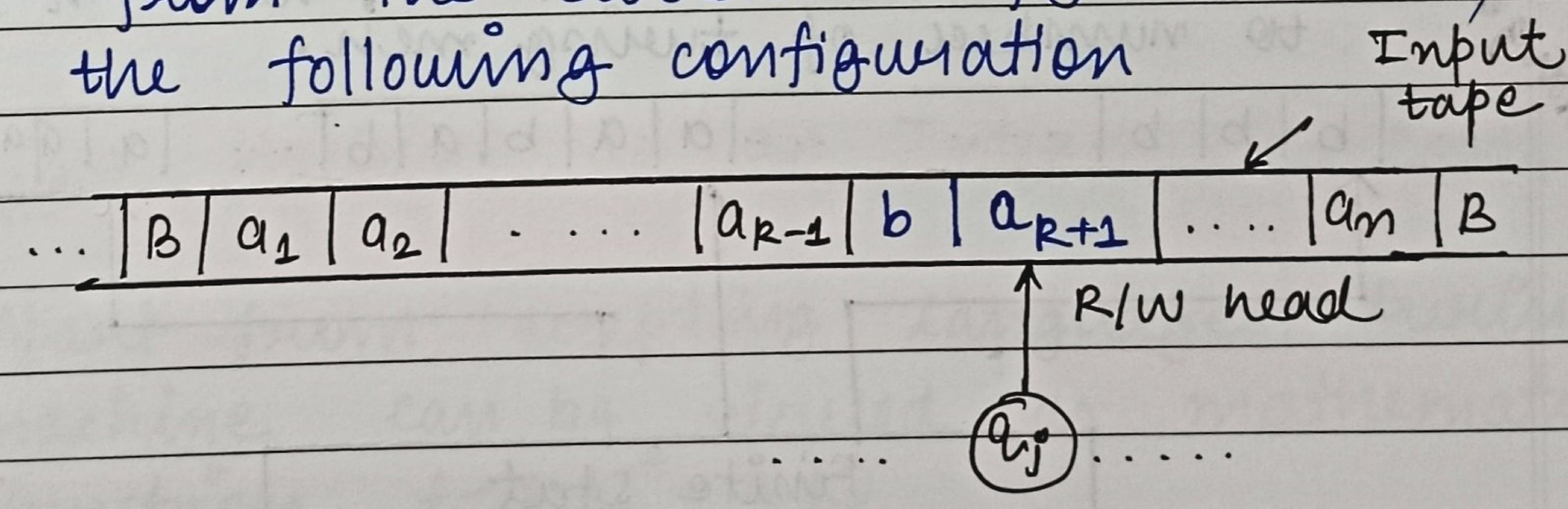
(ii) If the current configuration of TM is like



The ID will be written as

$$a_1 a_2 a_3 \dots a_{k-1} q_i a_k a_{k+1} a_{k+2} \dots a_n$$

(iii) If from the current configuration TM moves to the following configuration



i.e. $\delta(q_i, a_k) = (q_j, b, R)$

Then ID for transition will be written as

$$a_1 a_2 a_3 \dots a_{k-1} q_i a_k \dots a_n \vdash a_1 a_2 a_3 \dots a_{k-1} b q_j a_{k+1} \dots a_n$$

6 Modifications of Turing Machine

6.1 Turing Machines with stay operation

(i) New $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R, S\}$

(ii) Power of TM does not change

(iii) The stay operation S does not moves the R/W head

6.2 Multitape Turing M/c

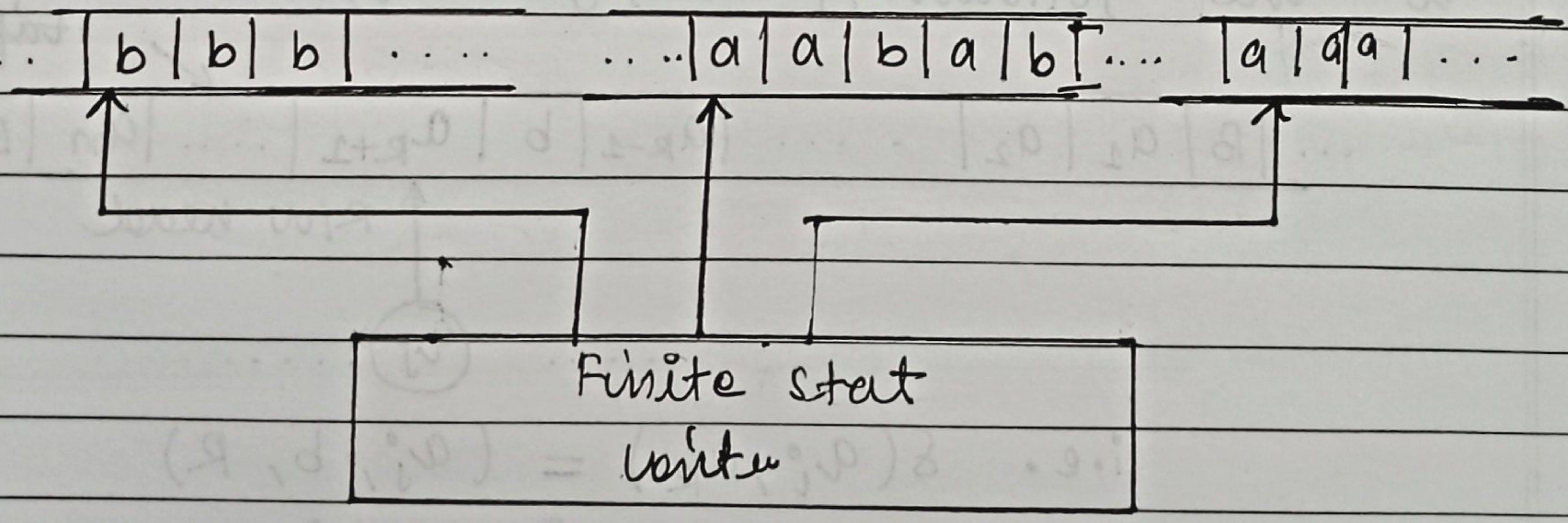
(i) In multitape turing machines ~~and~~ more than 1 tape exist

(ii) In multitape M/C transition function the transition function is defineds as

$$\delta : Q \times \Gamma^{a} \longrightarrow Q \times \Gamma^{a} \times \{L, R, S\}^{n}$$

Power of TM does not change

(iii) A formal view of multitape turing machine will be like, here n refers to number of turing mch.



6.3 Multidimensional Turing Machine

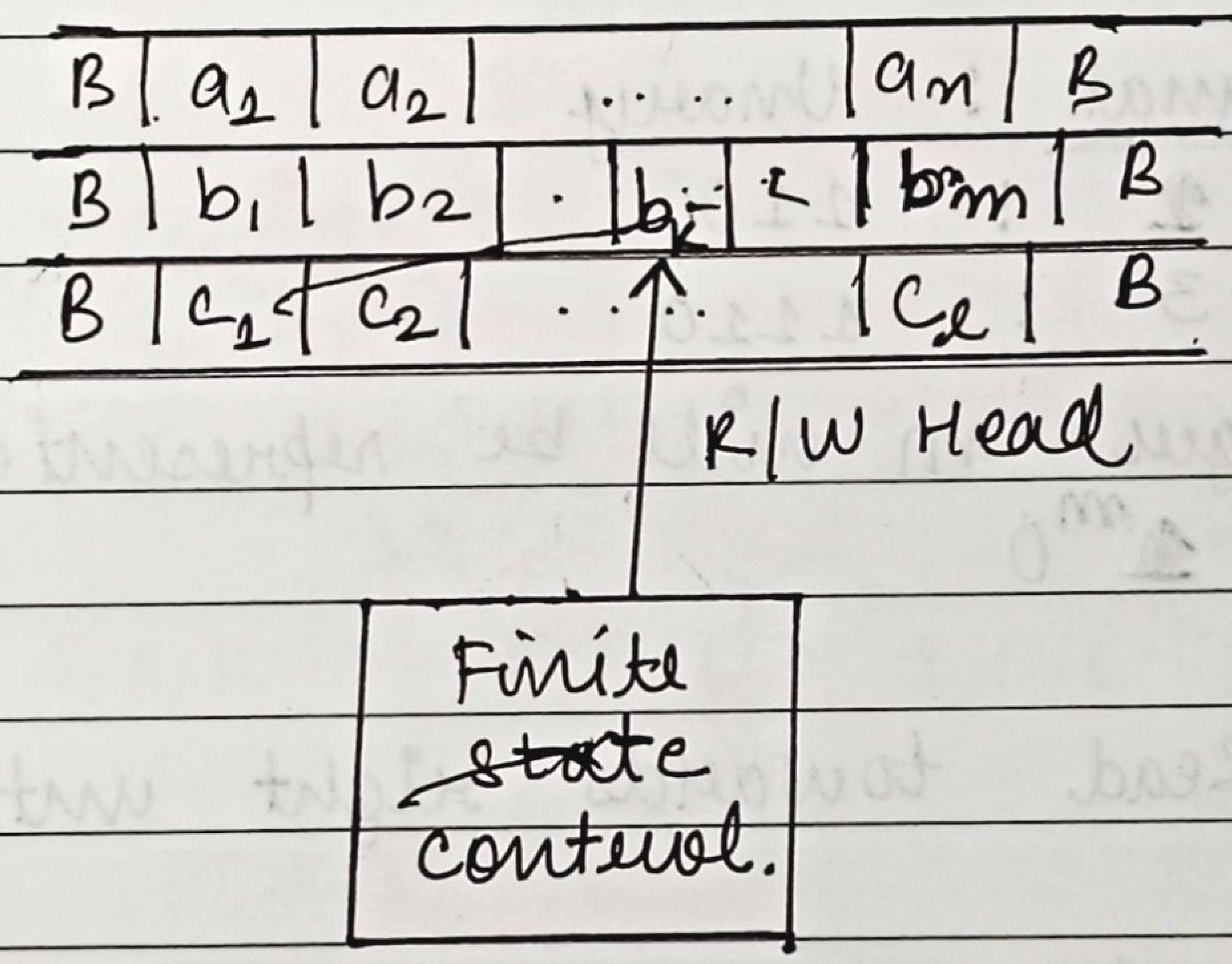
(i) In Multidimension TM. the input tape consist of multidimensional ~~state~~ rows and columns instead of linear tape.

(ii) The transition function for the multidimensional TM is

$$\delta : Q \times T \rightarrow Q \times T \times \{R, L, U, D \text{ or } S\}$$

(iii) A formal view of multidimensional (2-dimensional TM) will be like.

power of TM does not change



7 <u>Turing Machine as an Integer function</u>

(i) Apart from accepting languages, turing machine can be viewed as mathematical function $f : N^* \rightarrow N$

(ii) The input given is in encoded form usually binary or unary

Decimal Integer	Binary	Unary
2	10	11
4	100	1111
6	110	111111

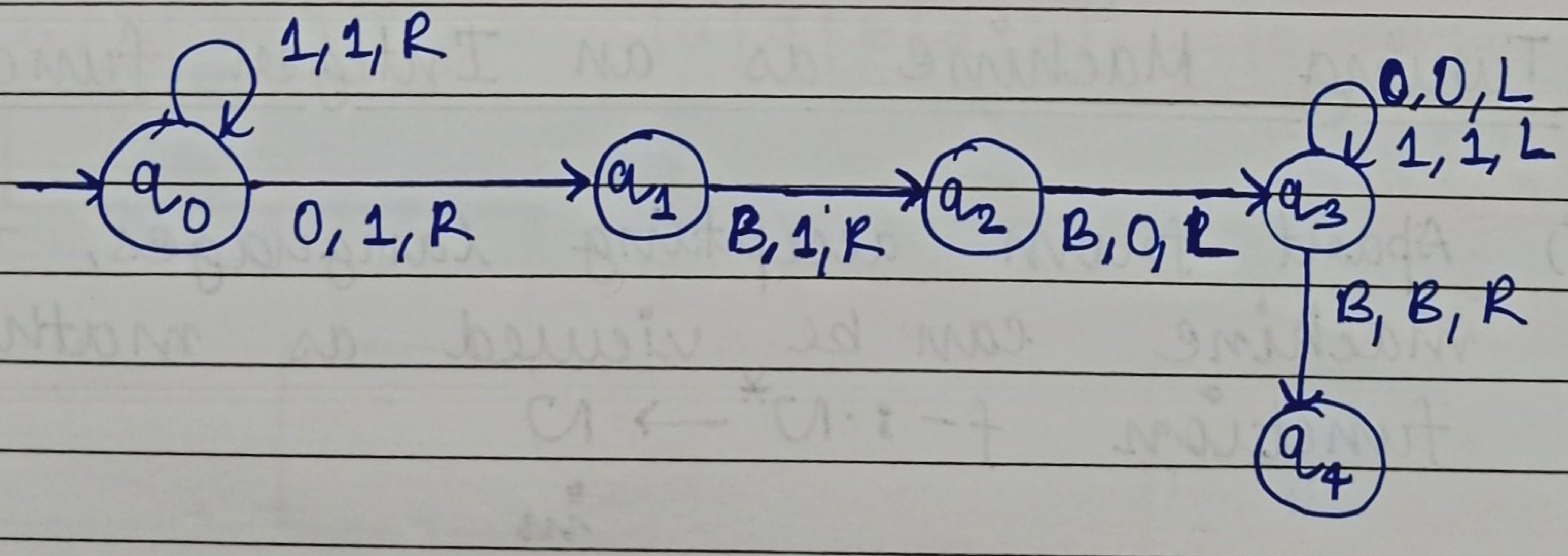
(iii) The output is whatever is remaining in the tape & the head will move to one direction while reading the output

Example : 1: Construct a turing machine that computes the function
$f(n) = n+2$

Input Format : Unary
As 2 : 110
3 : 1110
Any integer m will be represented as
$1^m 0$

Output : Read towards right until 0

Turing Machine

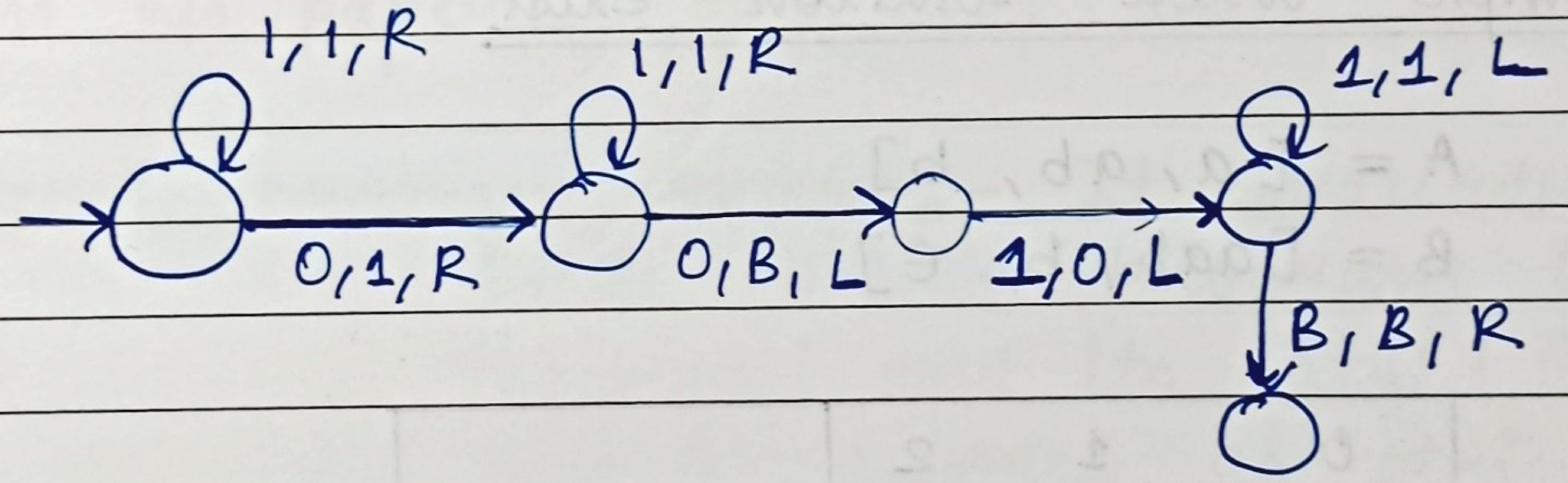


Example : 2 Construct a turing machine that computes the function
int f(a, b) {
return a+b
}

Input Format : Any number m will be represented in unary format as $1^m 0$

Output Format : Output will also be in unary format. Read head will read output towards right until 0

Turing Machine :



*** POST CORRESPONDANCE PROBLEM

1 Definition : PCP is a decision Problem where you are given two list of strings over the same alphabet. The goal is to find a sequence of indices such that the concatenated strings from both lists match

2 For List $A = [a_1, a_2, a_3 \ldots a_n]$
List $B = [b_1, b_2, b_3 \ldots b_n]$
find the indices $i_1, i_2, i_3 \ldots i_k$ such that
$a_{i_1} \cdot a_{i_2} \cdot a_{i_3} \cdot \ldots a_{i_k} = b_{i_1} \cdot b_{i_2} \cdot b_{i_3} \ldots b_{i_k}$

☞ PCP where indices are bound to start from first string of the list is called Modified PCP

3 Repetition of strings is allowed loop wise as well as chainwise
i.e
00 11 22 as well as 012012 0012

4 PCP is an undecidable problem for general inputs. No algorithm exists for arbitrary list that can always decides whether the solution exist or not

5 Example where solution exist

$A = [a, ab, b]$
$B = [aab, b, \epsilon]$

	0 1 2	
A	a · ab · b	= aabb
B	aab · b · ϵ	= aabb

Hence 0, 1, 2 is a valid solution

Example where solution does not exist

$A = [a, ab]$
$B = [b, a]$

Universal Turing Machine

1 **Definition :-** A universal turing machine is a general purpose theoretical turing machine. It takes as input the encoded description of a Turing Machine M and an input string w, and simulates the computation of M on w

2 It was invented by Alang Turing in 1936 in his paper " on computable Number"

3 **Stored program concept :-** It indroduces the idea that a program (Turing Machine) and its data (input) can be encoded as strings and given to single general purpose machine.

4 **Importance :-** UTM proves that there exist a single machine that can compute everything computable. It defines the limit of Algebraic computation.

5 **Disadvantage** It is a theoretical model and not practically implementable for large machines.