Question Nos.

Q2) b) Circular Linked List.

(1) Insertion at the Start.

```
Step 1 : Set ptr = Start

Step 2 : if ptr = NULL
            ptr = New node
            ~~New node = Start~~
         New node → next = Start.
            exit.

Step 3 : If Start != NULL.
```

Explaination.

~~Whe~~ we point the head to the new node. & then point the new nodes next ptr to the first node.

~~this way~~

Q2) b) Circular Linked List.

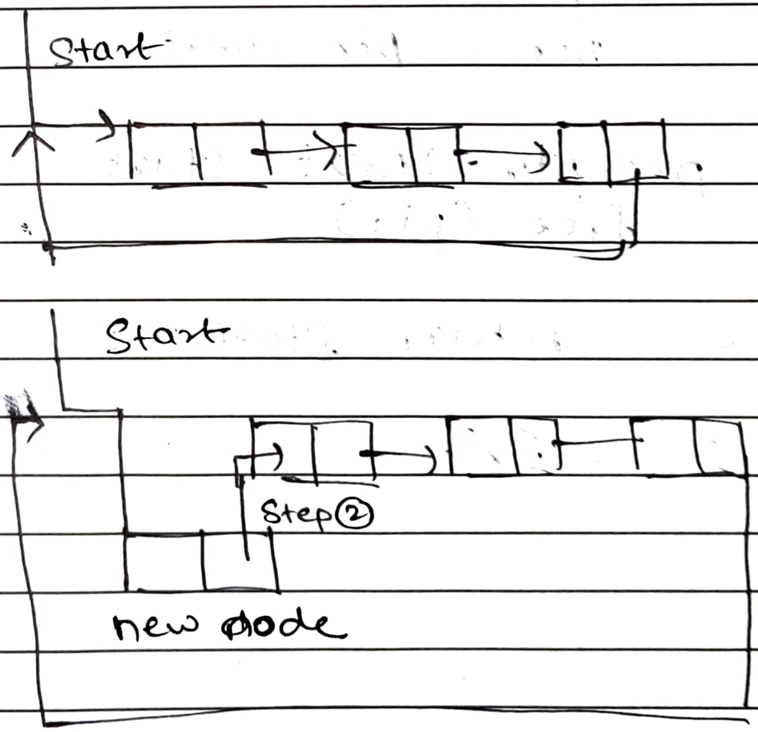
① Insertion at the start.

Step 1 : set ptr = ~~new~~ start (head)
input n = data.

Step 2 : set new node → data = n.
new node → next = start
~~8~~ ptr = new node.

Step 3 : ~~exit~~ return start
Step 4 : exit

Example.

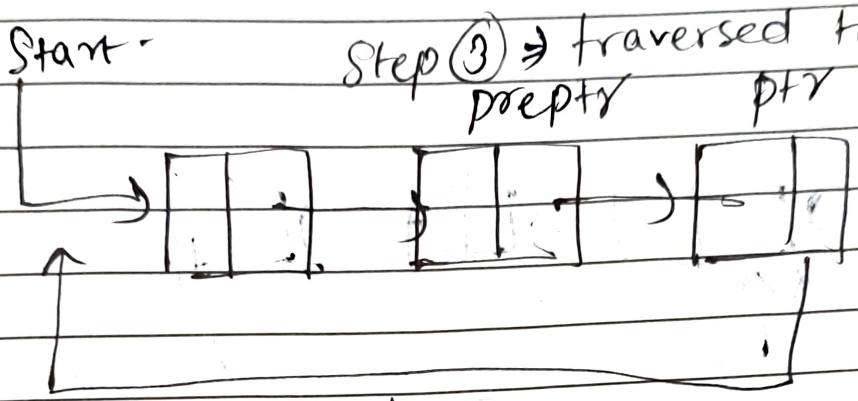


2.) Deletion of last node.

```
Step ①: Set ptr = ~~head~~ Start
            pre ptr = ptr
        input n = data.

Step ②: if ptr = NULL
        print ("List is Empty").
            Exit. Step 6.

Step ③: else ~~if~~ ~~ptr~~
        ~~do while (ptr != start)~~
        do while (ptr != start)
            preptr = ptr
            ptr = ptr → next

Step ④: pre ptr → next = Start
            free (ptr)

Step ⑤: return start.

Step ⑥: Exit.
```

Example.

Step ③ ⇒ traversed till the end.



we used do while (ptr != ~~Null~~ Start)
because there are two instances when ptr = Start once in the Start & once in the end. ~~&~~ what do while does is it atleast runs once so we pass the first instance when we got ptr = ~~a~~ start. next instance when ptr = start is at the end of the list. this when the do while loop breaks.

~~Step ④~~

we converge the prepointer to start & free the ptr which is at the end of the node. removing the last node from the list.