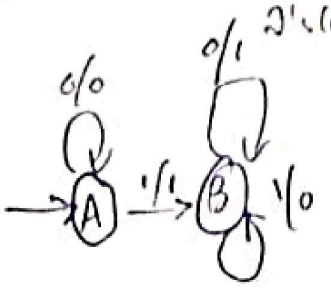
Mealy Machine 2's Complement

$\Sigma = \{0,1\}$ $\Delta = \{0,1\}$ eg:- 0 1 1 0 1 0 0 Reading Right to left.

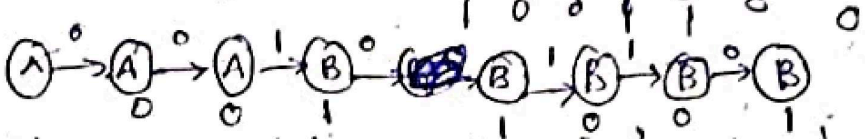
2's comp 1 0 0 1 1 0 0



	0	1
A	0	1
B	1	0

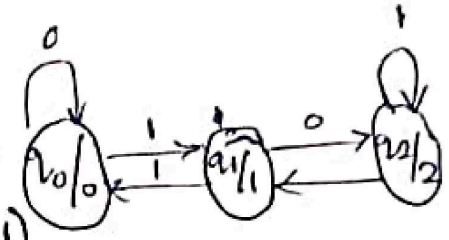
I/P Right to left 0 1 1 0 1 0 0

O/P Right to left 1 0 0 1 1 0 0

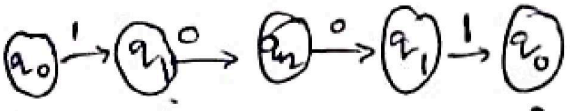


Moore's 3 Modulo :- $\Sigma = \{0,1\}$ $\Delta = \{0,1,2\}$

	0	1
q_0	q_0	q_1
q_1	q_2	q_0
q_2	q_1	q_2



(1 0 0 1)



0 1 2 1 0

(0 1 2 1 0)

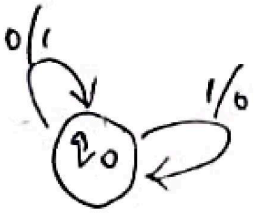
I/P L-R 1 0 0 1 1 0 0 (76)

O/P L-R 0 0 1 2 1 0 1 2 1 (1)

76 / 3 = 1 Hence Correct.

Mealy for 1's Complement :- $\Sigma = \{0,1\}$

$\Delta = \{0,1\}$



1 0 1 0 ← I/P

0 1 0 1 ← O/P

$q_0 \xrightarrow{1/0} q_0 \xrightarrow{0/1} q_0 \xrightarrow{1/0} q_0 \xrightarrow{0/1} q_0$

18B91A05C7 | Sriram

→ Write A program to implement following Double Linked List

(i) Insert End

(ii) Display Backward

(iii) Delete Begin.

Program-

```
#include<stdio.h>
#include <stdlib.h>

Struct dll {
        struct dll *prev;
        Struct dll *next;
        int data;
}

typedef struct dll* nptr;

nptr insert_back(nptr head, int x){
        nptr p, temp;
        p=(nptr)malloc(sizeof(nptr));
        p->data = x;
        p->next = NULL;
        temp= head;
        if (head == NULL){
            p->prev = NULL;
            return p;
        }
        while(temp->next! =NULL){
            temp=temp->next;
        }
        temp->next = p;
        p->prev = temp;
        return head;
}
```

```
Void disp (nptr head ){
        if (head == NULL){
            printf ("List is Empty");
            return;
        }
        printf ("\n Elements : ");
        nptr temp = head;
        while (temp -> next != NULL){
            temp = temp -> next;
        }
        while(temp != head){
            printf ("%d -> ", temp -> data);
            temp = temp -> prev;
        } printf("%d", head -> data);
}

nptr DeleteBegin (nptr head ){
        if(head == NULL){
            printf ("Invalid\n");
            return head;
        }
        nptr temp;
        temp = head;
        printf("%d is deleted\n", temp -> data);
        head = head -> next;
        head -> prev = NULL;
        free (temp);
        return head;
}
```

```
void main()
{
    node *head = NULL;
    int ch=0, x;
    while(ch != 4){
      printf("1. Insert End\n 2. Delete Begin\n
              3. Display Backward\n 4. Exit\n");
      printf("Enter Your Choice ");
      scanf("%d", &ch);
      switch(ch){
          case 1:
              printf("Enter Number: ");
              scanf("%d", &x);
              head = insert_back(head, x);
              break;
          case 2:
              printf("Deleting a Number\n");
              head = deleteBegin(head);
              break;
          case 3:
              printf("Displaying Backward\n");
              disp(head);
              break;
          case 4:
              printf("Exiting");
              break;
          default:
              printf("Enter a Valid option(1-4)\n");
      }
    }
}
```