Q) Write a C code to simulate linked list. insert element at the start, end, at position.

```
#include <stdio.h>
#include <stdlib.h>
Struct Node {
    int data;
    Struct Node * link;
};
typedef struct Node node;
 node * start = NULL;
Void create();
Void display();
Void Insertfromstart();
Void Insert at position();
Void Insert at end();
Void main() {
int ch;
while(1) {
 printf("\n1. create \n2 Display \n 3 Insert at
        beginning \n4 Insert at Position \n 5
        Insert at End \n6 Exit ");
 Scanf ("\n Enter your choice ");
print
Scanf ("%d ", &ch);
```

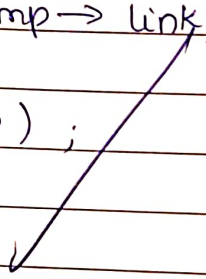
```
switch (ch) {
 case 1:
    create ();
    break ();
 case 2:
    display ();
    break;
 case 3:
   Insertatstart ();
    break;
 case 4:
   Insertatposition ();
    break;
case 5:
    Insert at End ();
    break;
case 6:
    exit (0);
 default :
   printf ("Enter a Number between 1 & 6 \n");
}
}
}
void create (){
char ch;
node *new1, *curr;
do{
   new1 = (node*) malloc (sizeof (node));
   printf ("\n enter value: \n");
    scanf ("%d", &new1->data);
   if (start == NULL)
    {
        start = new1;
        curr = new1; }
```

```
else {
curr -> link = new1;
curr = new1;
}
printf ("Do you want to add (Y/N)?");
scanf ("%c", &ch);
}
while (ch == 'Y' || ch == 'y');
curr -> link = NULL;
}

Void InsertfromStart () {
node new1 = (node) malloc (sizeof (node));
printf ("\n enter value : \n");
scanf ("%d", &new1 -> data);
if (start == NULL)
{
  start = new1;
  new1 -> link = NULL;
  return;
}
else {
  new1 -> link = start;
  start = new1;
  return;
}
}
```

```
int i=1; int pos;
node * temp=start;
printf("\n enter position:\n");
scanf("%d", &pos);
while (temp!= NULL && i< pos-1)
{
  temp= temp -> link;
  pos++;
}
if (temp= NULL)
   {
    pos++;
    return;
  }
 new1 -> link = temp -> link;
 temp -> link = new1;
}
Void InsertatEnd () {
  node new1= (node) malloc (sizeof (node));
   printf ("\n enter value \n");
   Scanf ("%d", &new1 -> data);
   if (start== NULL)
   {
   start = new1;
    new1 -> link = NULL;
    return;
}
node *  temp=start;
while (temp -> link!= NULL)
 {
  temp = temp -> link;
 }
 temp -> link = new1;
 new1 -> link = NULL; return; }
```

```
void display() {
  if (start == NULL) {
  printf ("\n Linked list is empty \n");
  return;
  }
 node * temp = start
 printf ("\n Linked list is empty \n");
  return ;
}
 node * temp = start;
 printf ("\n element in linked list ");
while (temp1 = NULL) {
  printf ("%d", temp->data);
  temp = temp-> link;
  }
  printf ("\n") ;
   }
```



Output

1) create
2) display
3) Insert at beginning
4) Insert at Position
5) Insert at end
6) Exit.
enter choice: 1
Enter value 5
do you want to continue. (Y/N) Y
enter choice: 2
5
do yo want to continue (Y/N) Y
enter choice 3
Enter value at start 4
do yo want to continue (Y/N) Y
enter choice 5
Enter value at end 6
do you want to continue (Y/N) Y
enter choice 2.
4 5 6
Enter choice : 4
Enter value : 10
Enter position : 2
Enter choice 2.
4 5 10 6.

26/11/24