NAME :- AINDRAIL SANTRA
ROLL NO :- 2029044

Q1. WAP to create a double linked list of n nodes and display the linked list by using suitable
user defined functions for create and display operations.

```
#include<stdio.h>
 #include<stdlib.h>
 struct Node
{ int data;
 struct Node* next;
 struct Node* prev; };
 void DinsertAtHead(struct Node** head, int data) { struct Node* node = (struct Node*)malloc(sizeof(struct Node));
node->data = data;
node->next = (*head);
 node->prev = NULL;
 if ((*head) != NULL)
 (*head)->prev = node;
(*head) = node;
 }
void DinsertAtTail(struct Node** head, int data)
{ struct Node* new_node = (struct Node*)malloc(sizeof(struct Node));
struct Node* last = *head;
```

```
  new_node->data = data;
new_node->next = NULL;
  if (*head == NULL)
  { new_node->prev = NULL;
  *head = new_node;
  return; }
while (last->next != NULL)
last = last->next;
last->next = new_node;
  new_node->prev = last; }
  void printDDList(struct Node* head)
{ while (head != NULL)
{ printf("%d ",head->data);
  head = head->next; }
printf("\n"); }
int main()
{ struct Node* head = NULL;
DinsertAtHead(&head,10);
DinsertAtHead(&head,11);
DinsertAtHead(&head,12);
  printDDList(head);
DinsertAtTail(&head,13);
DinsertAtTail(&head,14);
printDDList(head);
return 0;
}
```

OUTPUT

```
12 11 10
12 11 10 13 14
```

Q2. WAP to reverse the sequence elements in a double linked list.

```
#include<stdio.h>
#include<stdlib.h>
  struct Node {
  int data;
  struct Node* next;
  struct Node* prev;
};
void DinsertAtHead(struct Node** head, int
data)
{ struct Node* node = (struct
Node*)malloc(sizeof(struct Node));
node->data = data;
node->next = (*head);
  node->prev = NULL;
  if ((*head) != NULL)
  (*head)->prev = node;
(*head) = node;
}
void DinsertAtTail(struct Node** head, int
data)
{struct Node* new_node = (struct
Node*)malloc(sizeof(struct Node));
struct Node* last = *head;
new_node->data = data;
new_node->next = NULL;
if (*head == NULL) {
new_node->prev = NULL;
```

```
*head = new_node;
return;        }
while (last->next != NULL)
last = last->next;
last->next = new_node;
new_node->prev = last;
} void printDDList(struct Node* head)
{      printf("The doubly linked list is : ");
while (head != NULL)
{printf("%d ",head->data);
head = head->next;        }
printf("\n"); }
void reverseDD( struct Node** head)
{struct Node *temp = NULL;
struct Node *current = *head;
while (current !=  NULL)          {
temp = current->prev;
current->prev = current->next;
current->next = temp;
current = current->prev;          }
if(temp != NULL )
*head = temp->prev; }
int main(){
struct Node* head = NULL;
int n, num;
printf("Enter the value of n \n");
scanf("%d", &n);
for( int i = 0 ; i<n; i++)
```

```
{scanf("%d",&num);
DinsertAtTail(&head,num);          }
printf("ORIGINAL\n");
printDDList(head);
reverseDD(&head);
printf("Reversed\n");
printDDList(head);
return 0; }
```

OUTPUT

```
Enter the value of n

10

1 2 3 4 5 6 7 8 9 10

ORIGINAL

The doubly linked list is : 1 2 3 4 5 6 7
8 9 10

Reversed

The doubly linked list is : 10 9 8 7 6 5 4
3 2 1
```