DS PRACTICAL 07

AIM :- Implement a Circular Single Linked List and Perform the operation : Create, Traverse, Insert_beg, Insert_end, Delete_beg, Delete_end using Menu Driver Program.

PROGRAM :-

```
#include<stdio.h>
#include<stdlib.h>
struct node{
   int data;
   struct node*next;
};
struct node *s, *p, *q, *a, *t;

//Function to create a circular linked list
void create(){
   printf("Creating the Circular Linked List(CLL).\nEnter data for the First node:\t");
   p = (struct node *)malloc(sizeof(struct node));
   scanf("%d", &p -> data);
   p -> next = p;
   s = p;
}

//Function to traverse through circular linked list
void Traverse(){
   printf("\nTraversing the linked list:\t");
   t = s;
   do{
      printf("%d\t", t -> data);
      t = t -> next;
   } while (t != s);
}

// Function to Insert at beggining through circular linked list
```

```
void Insert_Beg()
{   printf("\nInserting node at beggining.\nEnter the data:\t");
    p = (struct node *)malloc(sizeof(struct node));
    scanf("%d", &(p -> data));

    if(s == NULL){
        p -> next = p;
        s = p;
    }else{
        t = s;
        while (t -> next != s)
        {
            t = t-> next;
        }
        p->next = s;
        t->next = p;
        s = p;
    }


}

// Function to Insert at end through circular linked list
void Insert_End()
{
    t =s;
    while(t -> next != s){
        t = t -> next;
    }

    p = (struct node*)malloc(sizeof(struct node));
```

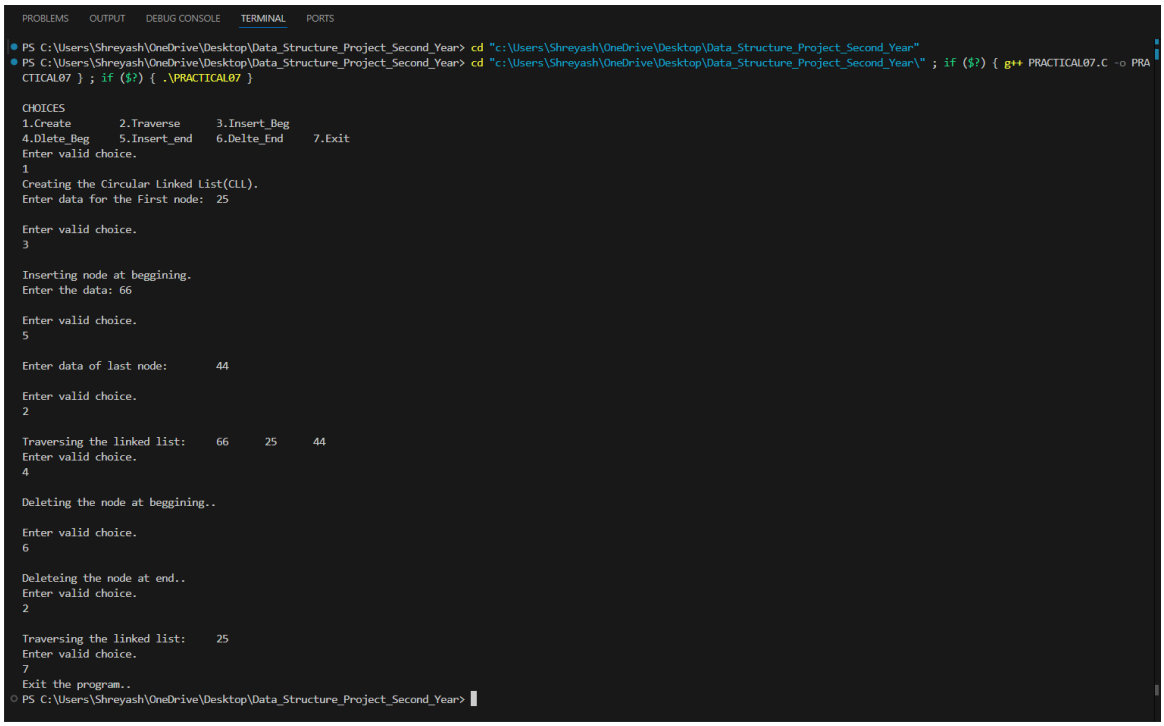
```
    printf("\nEnter data of last node:\t");
    scanf("%d", &(p -> data));
    p -> next = s;
    t -> next =  p;
}

// Function to Delete at beggining through circular linked list
void Delete_Beg()
{   printf("\nDeleting the node at beggining..\n");
if (s == NULL)
{
   printf("The linked list is empty..\n");
}

    t = s;
    while(t -> next != s){
        t = t-> next;
    }
    q = s -> next;
    t -> next = q;
    free(s);
    s = q;

}

// Function to Delete at end through circular linked list
void Delete_End()
{   printf("\nDeleteing the node at end..");
    t = s;
    while(t -> next != s){
        q = t;
```

```
        t = t-> next;
    }
    q -> next = s;
    free(t);
}
int main(){
    int choice;

    printf("\nCHOICES\n1.Create\t2.Traverse\t3.Insert_Beg\n4.Dlete_Beg\t5.Insert_end\t6.Delte_End\
t7.Exit");
    do{
        printf("\nEnter valid choice.\n");
        scanf("%d", &choice);
        switch (choice)
        {
        case 1:
            create();
            break;

        case 2:
            Traverse();
            break;

        case 3:
            Insert_Beg();
            break;

        case 4:
            Delete_Beg();
            break;
```

```
        case 5:
            Insert_End();
            break;

        case 6:
            Delete_End();
            break;

        case 7:
            printf("Exit the program..");
            break;

        default:
        printf("\nPlease enter a valid choice\n");
            break;
        }
    }while(choice != 7);

}
```

DS PRACTICAL 07

OUTPUT



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\Shreyash\OneDrive\Desktop\Data_Structure_Project_Second_Year> cd "c:\Users\Shreyash\OneDrive\Desktop\Data_Structure_Project_Second_Year"
PS C:\Users\Shreyash\OneDrive\Desktop\Data_Structure_Project_Second_Year> cd "c:\Users\Shreyash\OneDrive\Desktop\Data_Structure_Project_Second_Year\" ; if ($?) { g++ PRACTICAL07.C -o PRA
CTICAL07 } ; if ($?) { .\PRACTICAL07 }

CHOICES
1.Create        2.Traverse      3.Insert_Beg
4.Dlete_Beg     5.Insert_end    6.Delte_End     7.Exit
Enter valid choice.
1
Creating the Circular Linked List(CLL).
Enter data for the First node:  25

Enter valid choice.
3

Inserting node at beggining.
Enter the data: 66

Enter valid choice.
5

Enter data of last node:        44

Enter valid choice.
2

Traversing the linked list:     66      25      44
Enter valid choice.
4

Deleting the node at beggining..

Enter valid choice.
6

Deleteing the node at end..
Enter valid choice.
2

Traversing the linked list:     25
Enter valid choice.
7
Exit the program..
PS C:\Users\Shreyash\OneDrive\Desktop\Data_Structure_Project_Second_Year>
```

GIT-HUB LINK :- https://github.com/ShreyashGajbhiye453/Data-Structure-Practical-No.-01