**DSA Assignment: 8**

**Exp 8:** Implementation of Circular Doubly Linked List

Shashwat Tripathi

D10A Roll No: 60

**AIM:** In this experiment, we will implement circular doubly linked list.

**CODE:**

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
    struct node *prev;
    struct node *next;
    int data;
};
struct node *head;
void insertion_beginning();
void insertion_last();
void deletion_beginning();
void deletion_last();
void display();
void search();
void main()
{
    printf("D10A_60_ShashwatTripathi\n");
    printf("\n###############################################");
    printf("\n1.Insert in Beginning");
    printf("\n2.Insert at end");
    printf("\n3.Delete from beginning");
    printf("\n4.Delete from end");
    printf("\n5.Search");
    printf("\n6.Show");
    printf("\n7.Exit");
    printf("\n###############################################");

    int choice = 0;
    while (choice != 9)
    {
        printf("\nEnter your choice:");
        scanf("\n%d", &choice);
        switch (choice)
        {
        case 1:
            insertion_beginning();
            break;
        case 2:
            insertion_last();
            break;
        case 3:
            deletion_beginning();
            break;
        case 4:
```

```
            deletion_last();
            break;
        case 5:
            search();
            break;
        case 6:
            display();
            break;
        case 7:
            exit(0);
            break;
        default:
            printf("Please enter valid choice..");
        }
    }
}
void insertion_beginning()
{
    struct node *ptr, *temp;
    int item;
    ptr = (struct node *)malloc(sizeof(struct node));
    if (ptr == NULL)
    {
        printf("\nOVERFLOW\n");
    }
    else
    {
        printf("\nEnter Item value: ");
        scanf("%d", &item);
        ptr->data = item;
        if (head == NULL)
        {
            head = ptr;
            ptr->next = head;
            ptr->prev = head;
        }
        else
        {
            temp = head;
            while (temp->next != head)
            {
                temp = temp->next;
            }
            temp->next = ptr;
            ptr->prev = temp;
            head->prev = ptr;
            ptr->next = head;
            head = ptr;
        }
        printf("\nNode inserted\n");
    }
}
void insertion_last()
{
    struct node *ptr, *temp;
    int item;
    ptr = (struct node *)malloc(sizeof(struct node));
```

```
    if (ptr == NULL)
    {
        printf("\nOVERFLOW");
    }
    else
    {
        printf("\nEnter value: ");
        scanf("%d", &item);
        ptr->data = item;
        if (head == NULL)
        {
            head = ptr;
            ptr->next = head;
            ptr->prev = head;
        }
        else
        {
            temp = head;
            while (temp->next != head)
            {
                temp = temp->next;
            }
            temp->next = ptr;
            ptr->prev = temp;
            head->prev = ptr;
            ptr->next = head;
        }
    }
    printf("\nnode inserted\n");
}

void deletion_beginning()
{
    struct node *temp;
    if (head == NULL)
    {
        printf("\n UNDERFLOW");
    }
    else if (head->next == head)
    {
        head = NULL;
        free(head);
        printf("\nnode deleted\n");
    }
    else
    {
        temp = head;
        while (temp->next != head)
        {
            temp = temp->next;
        }
        temp->next = head->next;
        head->next->prev = temp;
        free(head);
        head = temp->next;
    }
}
```

```c
void deletion_last()
{
    struct node *ptr;
    if (head == NULL)
    {
        printf("\n UNDERFLOW");
    }
    else if (head->next == head)
    {
        head = NULL;
        free(head);
        printf("\nnode deleted\n");
    }
    else
    {
        ptr = head;
        if (ptr->next != head)
        {
            ptr = ptr->next;
        }
        ptr->prev->next = head;
        head->prev = ptr->prev;
        free(ptr);
        printf("\nnode deleted\n");
    }
}

void display()
{
    struct node *ptr;
    ptr = head;
    if (head == NULL)
    {
        printf("\nnothing to print");
    }
    else
    {
        printf("\n Printing the list ... \n");

        while (ptr->next != head)
        {

            printf("%d\n", ptr->data);
            ptr = ptr->next;
        }
        printf("%d\n", ptr->data);
    }
}

void search()
{
    struct node *ptr;
    int item, i = 0, flag = 1;
    ptr = head;
    if (ptr == NULL)
    {
        printf("\nEmpty List\n");
```

```
        }
        else
        {
            printf("\nEnter item which you want to search?: \n");
            scanf("%d", &item);
            if (head->data == item)
            {
                printf("item found at location: %d", i + 1);
                flag = 0;
            }
            else
            {
                while (ptr->next != head)
                {
                    if (ptr->data == item)
                    {
                        printf("item found at location: %d ", i + 1);
                        flag = 0;
                        break;
                    }
                    else
                    {
                        flag = 1;
                    }
                    i++;
                    ptr = ptr->next;
                }
            }
            if (flag != 0)
            {
                printf("Item not found\n");
            }
        }
    }
```

**OUTPUT:**

C:\Windows\System32\cmd.exe

```
C:\Users\shweta\Documents\Shashwat\Notepad++\DSA>DSAexp8
D10A_60_ShashwatTripathi

###########################################
1.Insert in Beginning
2.Insert at end
3.Delete from beginning
4.Delete from end
5.Search
6.Show
7.Exit
###########################################
Enter your choice:1

Enter Item value: 20

Node inserted

Enter your choice:1

Enter Item value: 25

Node inserted

Enter your choice:2

Enter value: 30

node inserted

Enter your choice:6

 Printing the list ...
25
20
30
```

C:\Windows\System32\cmd.exe

```
 Printing the list ...
25
20
30

Enter your choice:3

Enter your choice:4

node deleted

Enter your choice:6

 Printing the list ...
20

Enter your choice:5

Enter item which you want to search?:
45
Item not found

Enter your choice:7

C:\Users\shweta\Documents\Shashwat\Notepad++\DSA>
```