Page No.
Date

125
132
DSL

Experiment No 6

Aim :- Implementation of Doubly linked list and various operation for real-world applications.

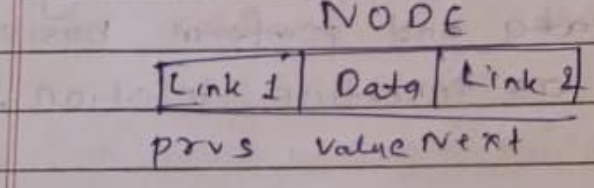
Objectives:

To use the basic principle of programming as applied to complex data structures.
To learn the principles of doubly linked lists and its various operations.

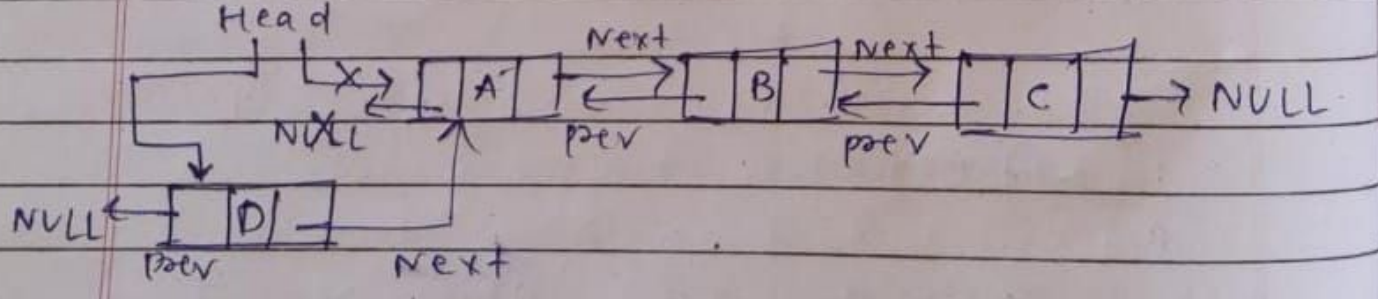
Theory :-

Introduction to doubly linked list :-

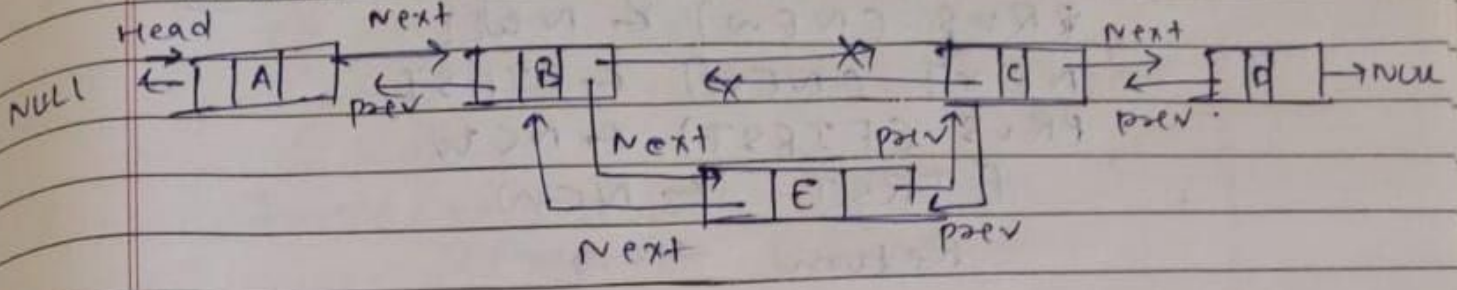
In doubly linked list every node has link to its previous and next node so that, we can traverse forward and backward. Every node in doubly linked list contains 3 fields.



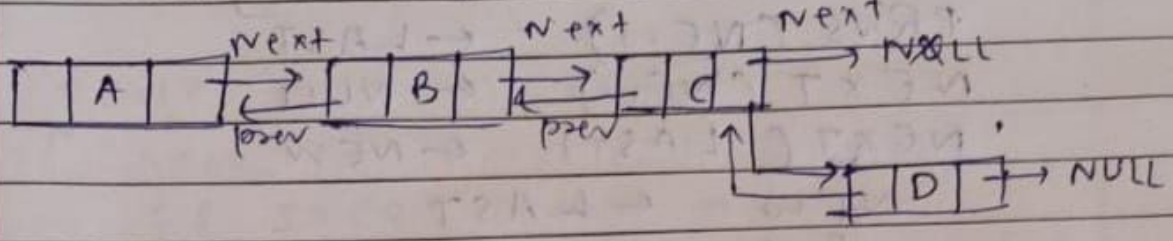
Insert at Beginning



Insert at middle



Insert at end



Algorithm:-

1. Insertion in doubly linked list
procedure DOUBINS(FIRST, LAST, X, VAL)
given doubly linked list leftmost and rightmost node as FIRST and LAST resp.
Insert node named NEW, left and right links of node by PRVS and NEXT
information in NEW node is X

1. [obtain new node ~~in~~ from stack]
NEW ← NODE
2. [copy information field]
INFO(NEW) ← X
3. [Insertion into an empty list]
if FIRST = LAST = NULL
PRVS(NEW) ← NEXT(NEW) ← NULL
FIRST ← LAST ← NULL
Return

4. [Insertion at beginning of list]
PRVS (NEW) ← NULL
NEXT (NEW) ← FIRST
PRVS (FIRST) ← NEW
FIRST ← NEW
Return

5. [Insertion at end of list]
PRVS (NEW) ← LAST
NEXT (NEW) ← NULL
NEXT (LAST) ← NEW
NEW ← LAST
Return

6. [Insertion after node whose value is VAL]
TEMP → FIRST
WHILE (INFO (TEMP)) <= VAL
TEMP ← NEXT (TEMP)
PRVS (NEXT (TEMP)) ← NEW
NEXT (TEMP) ← NEW
Return

(2) Deletion in Doubly linked list

Procedure DOUBDEL (FIRST, LAST, X)
given doubly linked list of leftmost and rightmost node as FIRST and LAST help Delete node whose val is X, left and right links of node by PRVS and NE

1. [Check if list is empty]
If FIRST = LAST = NULL
then write ('Empty List')
Return

2. [Initialize search for X)
TEMP ← FIRST

3. [FIND X]
While INFO (TEMP) ≠ X and NEXT (TEMP) ≠ NULL
TEMP ← NEXT (TEMP)

4. [Check if it is end of list]
if INFO (TEMP) ≠ X
then write ('Node not found')
Return

5. [Return deleted node to availability area]
NEXT (TEMP) ← AVAIL
AVAIL ← TEMP
Return

Conclusion :- Able to learn the implementation of doubly linked list through the understanding the code and its performance / output.

Outcome :- Able to apply the concepts of singly, circular and doubly linked list for real world applications.

doublyLL - Notepad

File Edit Format View Help

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

struct node
{
    struct node *prev;
    int data;
    struct node *next;
} * head, *temp, *temp1, *temp2;

void insert_beg();
void insert_end();
void insert_mid();
void delete ();
void display_forward();
void display_backward(int i);

int count = 0;

void main()
{
    int choice, insert_option, print_option;
    printf("Welcome to the implementation of the doubly linked list ! \n");
    do
    {
        printf("\n Please select an operation to perform from the below list \n");
        printf(" 1. Insert a node \n 2. Delete a node \n 3. Print the existing list \n 4. Exit \n");
        printf("Enter your choice: ");
        scanf("%d", &choice);
        printf("\n \n");
        switch (choice)
        {
        case 1:
            do
            {
                printf("Select a position where you to want to insert new node \n");
                printf(" 1. Beginning of the List \n 2. At the end of the list \n 3. Insert in between \n 4. Exit the insert operation \n");
                printf("Enter your choice: ");
                scanf("%d", &insert_option);
                switch (insert_option)
                {
```

C:\Windows\system32\cmd.exe - a

```
Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lenov>cd desktop

C:\Users\lenov\Desktop>cd cmd

C:\Users\lenov\Desktop\cmd>gcc Doublyll.c

C:\Users\lenov\Desktop\cmd>a
Welcome to the implementation of the doubly linked list !

 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 1


Select a position where you to want to insert new node
 1. Beginning of the List
 2. At the end of the list
 3. Insert in between
 4. Exit the insert operation
Enter your choice: 2
Enter the data to be inserted: 2

Select a position where you to want to insert new node
 1. Beginning of the List
```

Ln 39, Col 47 | 100% | Windows (CRLF) | UTF-8

doublyLL - Notepad

File Edit Format View Help

```
        } while (insert_option != 4);
        printf("\n \n");
        break;
    case 2:
        delete ();
        break;
    case 3:
        do
        {
            printf(" --Display Option Menu-- \n");
            printf(" 1. Print List in Forward Direction \t2. Print List in Back
            printf(" Enter your choice: ");
            scanf("%d", &print_option);
            switch (print_option)
            {
            case 1:
                display_forward();
                printf("\n \n");
                break;
            case 2:
                temp2 = head;
                if (temp2 == NULL)
                    printf(" Error : List empty to display \n");
                else
                {
                    printf(" Linked list elements in backward direction : ");
                    display_backward(temp2->data);
                }
                printf("\n \n");
                break;
            case 3:
                printf(" Print Operation Exit !! \n");
                break;
            default:
                printf(" Please enter a valid option: 1, 2, 3 \n");
                break;
            }
        } while (print_option != 3);
        break;
    case 4:
        printf("Exit: Program Finished !!");
        break;
```

C:\Windows\system32\cmd.exe - a

```
Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lenov>cd desktop

C:\Users\lenov\Desktop>cd cmd

C:\Users\lenov\Desktop\cmd>gcc Doublyll.c

C:\Users\lenov\Desktop\cmd>a
Welcome to the implementation of the doubly linked list !

 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 1


Select a position where you to want to insert new node
 1. Beginning of the List
 2. At the end of the list
 3. Insert in between
 4. Exit the insert operation
Enter your choice: 2
Enter the data to be inserted: 2

Select a position where you to want to insert new node
 1. Beginning of the List
```

Ln 39, Col 47 | 100% | Windows (CRLF) | UTF-8

doublyLL - Notepad

ile Edit Format View Help

```
/ Function to insert element
oid insert_beg()

   if (head == NULL)
   {
       create();
       head = temp;
       temp1 = head;
   }
   else
   {
       create();
       temp->next = head;
       head->prev = temp;
       head = temp;
   }

oid insert_end()

   if (head == NULL)
   {
       create();
       head = temp;
       temp1 = head;
   }
   else
   {
       create();
       temp1->next = temp;
       temp->prev = temp1;
       temp1 = temp;
   }

oid insert_mid()

   int pos, i = 2;

   printf(" Enter position of the element to be inserted : ");
   scanf("%d", &pos);
   temp2 = head;
```

C:\Windows\system32\cmd.exe - a

```
Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lenov>cd desktop

C:\Users\lenov\Desktop>cd cmd

C:\Users\lenov\Desktop\cmd>gcc Doublyll.c

C:\Users\lenov\Desktop\cmd>a
Welcome to the implementation of the doubly linked list !

 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 1


Select a position where you to want to insert new node
 1. Beginning of the List
 2. At the end of the list
 3. Insert in between
 4. Exit the insert operation
Enter your choice: 2
Enter the data to be inserted: 2

Select a position where you to want to insert new node
 1. Beginning of the List
```

Ln 39, Col 47 | 100% | Windows (CRLF) | UTF-8

doublyLL - Notepad

File Edit Format View Help

```
    if ((head == NULL) && (pos != 1))
    {
        printf("\n Empty list cannot insert other than 1st position");
        return;
    }
    if ((head == NULL) && (pos == 1))
    {
        create();
        head = temp;
        temp1 = head;
        return;
    }
    else
    {
        while (i < pos)
        {
            temp2 = temp2->next;
            i++;
        }
        create();
        temp->prev = temp2;
        temp->next = temp2->next;
        temp2->next->prev = temp;
        temp2->next = temp;
    }
}

// Function to delete element
void delete ()
{
    int i = 1, pos;

    printf("\n Enter position of the element to be deleted : ");
    scanf("%d", &pos);
    temp2 = head;

    if ((pos < 1) || (pos >= count + 1))
    {
        printf(" Error : Position out of range to delete \n");
        return;
    }
    if (head == NULL)
```

C:\Windows\system32\cmd.exe - a

```
Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lenov>cd desktop

C:\Users\lenov\Desktop>cd cmd

C:\Users\lenov\Desktop\cmd>gcc Doublyll.c

C:\Users\lenov\Desktop\cmd>a
Welcome to the implementation of the doubly linked list !

 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 1


Select a position where you to want to insert new node
 1. Beginning of the List
 2. At the end of the list
 3. Insert in between
 4. Exit the insert operation
Enter your choice: 2
Enter the data to be inserted: 2

Select a position where you to want to insert new node
 1. Beginning of the List
```

Ln 39, Col 47 | 100% | Windows (CRLF) | UTF-8

doublyLL - Notepad

File Edit Format View Help

```
            return;
        }
        temp2->next->prev = temp2->prev;
        if (i != 1)
            temp2->prev->next = temp2->next;
        if (i == 1)
            head = temp2->next;
        printf(" Node deleted \n");
        free(temp2);
    }
    count--;
}

// Function to display elements
void display_forward()
{
    temp2 = head;

    if (temp2 == NULL)
    {
        printf("List empty to display \n");
        return;
    }
    printf(" Linked list elements in forward direction : ");

    while (temp2->next != NULL)
    {
        printf(" %d ", temp2->data);
        temp2 = temp2->next;
    }
    printf(" %d ", temp2->data);
}
void display_backward(int i)
{
    if (temp2 != NULL)
    {
        i = temp2->data;
        temp2 = temp2->next;
        display_backward(i);
        printf(" %d ", i);
    }
}
```

C:\Windows\system32\cmd.exe - a

```
Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lenov>cd desktop

C:\Users\lenov\Desktop>cd cmd

C:\Users\lenov\Desktop\cmd>gcc Doublyll.c

C:\Users\lenov\Desktop\cmd>a
Welcome to the implementation of the doubly linked list !

 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 1


Select a position where you to want to insert new node
 1. Beginning of the List
 2. At the end of the list
 3. Insert in between
 4. Exit the insert operation
Enter your choice: 2
Enter the data to be inserted: 2

Select a position where you to want to insert new node
 1. Beginning of the List
```

Ln 39, Col 47 | 100% | Windows (CRLF) | UTF-8

```
C:\Windows\system32\cmd.exe - a
icrosoft Windows [Version 10.0.19042.1348]
c) Microsoft Corporation. All rights reserved.

:\Users\lenov>cd desktop

:\Users\lenov\Desktop>cd cmd

:\Users\lenov\Desktop\cmd>gcc Doublyll.c

:\Users\lenov\Desktop\cmd>a
elcome to the implementation of the doubly linked list !

Please select an operation to perform from the below list
1. Insert a node
2. Delete a node
3. Print the existing list
4. Exit
nter your choice: 1


elect a position where you to want to insert new node
1. Beginning of the List
2. At the end of the list
3. Insert in between
4. Exit the insert operation
nter your choice: 2
nter the data to be inserted: 2

elect a position where you to want to insert new node
1. Beginning of the List
2. At the end of the list
3. Insert in between
4. Exit the insert operation
nter your choice: 2
nter the data to be inserted: 3

elect a position where you to want to insert new node
1. Beginning of the List
2. At the end of the list
3. Insert in between
4. Exit the insert operation
nter your choice: 2
nter the data to be inserted: 5

elect a position where you to want to insert new node
1. Beginning of the List
2. At the end of the list
3. Insert in between
4. Exit the insert operation
nter your choice: 3
```

```
C:\Windows\system32\cmd.exe - a
1. Beginning of the List
2. At the end of the list
3. Insert in between
4. Exit the insert operation
Enter your choice: 3
 Enter position of the element to be inserted : 2
Enter the data to be inserted: 4

Select a position where you to want to insert new node
 1. Beginning of the List
 2. At the end of the list
 3. Insert in between
 4. Exit the insert operation
Enter your choice: 4
Insert operation Exit


 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 3


 --Display Option Menu--
 1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
 Enter your choice: 1
 Linked list elements in forward direction :  2  4  3  5

 --Display Option Menu--
 1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
 Enter your choice: 2
 Linked list elements in backward direction :  5  3  4  2

 --Display Option Menu--
 1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
 Enter your choice: 3
 Print Operation Exit !!

 Please select an operation to perform from the below list
 1. Insert a node
 2. Delete a node
 3. Print the existing list
 4. Exit
Enter your choice: 2
```

C:\Windows\system32\cmd.exe - a

```
--Display Option Menu--
1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
Enter your choice: 2
Linked list elements in backward direction :  5  3  4  2

--Display Option Menu--
1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
Enter your choice: 3
Print Operation Exit !!

Please select an operation to perform from the below list
1. Insert a node
2. Delete a node
3. Print the existing list
4. Exit
Enter your choice: 2



Enter position of the element to be deleted : 2
Node deleted

Please select an operation to perform from the below list
1. Insert a node
2. Delete a node
3. Print the existing list
4. Exit
Enter your choice: 3


--Display Option Menu--
1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
Enter your choice: 1
Linked list elements in forward direction :  2  3  5

--Display Option Menu--
1. Print List in Forward Direction     2. Print List in Backward Direction     3. Exit
Enter your choice: 3
Print Operation Exit !!

Please select an operation to perform from the below list
1. Insert a node
2. Delete a node
3. Print the existing list
4. Exit
Enter your choice:
```