

1. Traverse

```
#include <stdio.h>

#include <stdlib.h>

struct node
{
    int data;
    struct node *next;
};

int main()
{
    struct node *s, *p, *q;
    int ch;
    p = (struct node *)malloc(sizeof(struct node));
    printf("Enter data of first node: ");
    scanf("%d", &p->data);
    s = p;
    do
    {
        q = (struct node *)malloc(sizeof(struct node));
        printf("\nEnter data of next node: ");
        scanf("%d", &q->data);
        p->next = q;
        p = q;
        printf("press 1 to add next node: ");
        scanf("%d", &ch);
    }
    ;
```

```
} while (ch == 1);

p->next = NULL;

printf("\nLinked list: ");

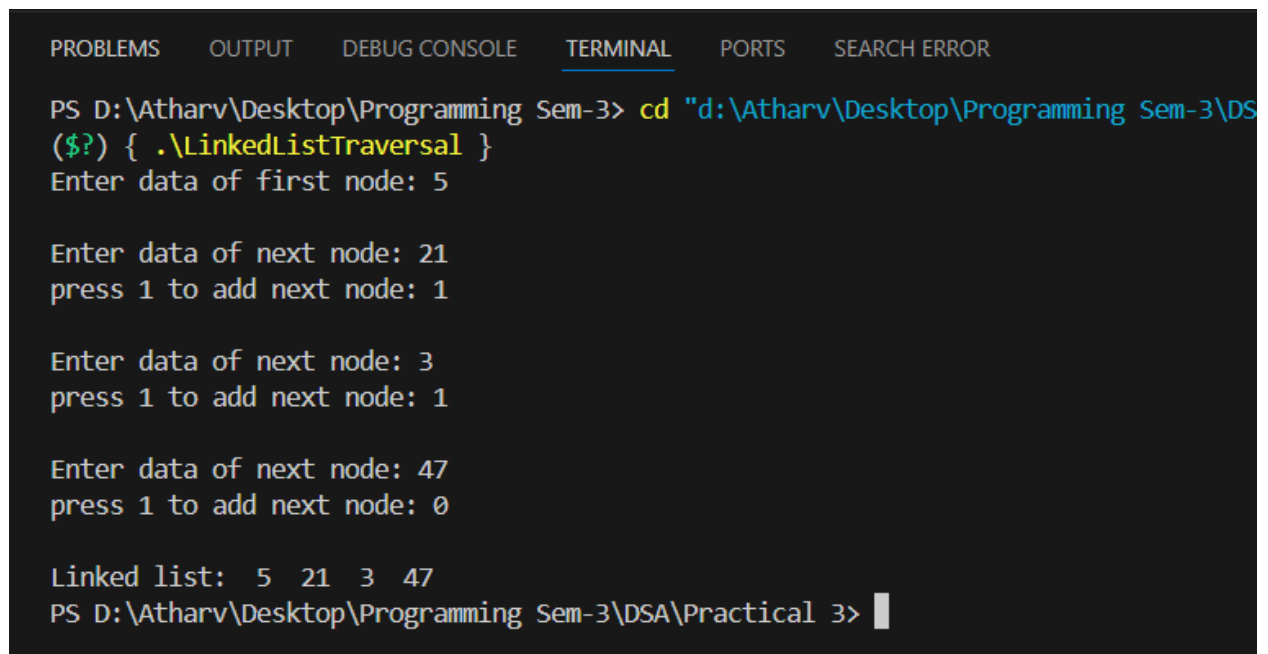
while (s != NULL)
{

    printf(" %d ", s->data);

    s = s->next;

}

}
```



The screenshot shows a Windows terminal window with a dark background. At the top, there are tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is active), 'PORTS', and 'SEARCH ERROR'. The terminal text shows the user navigating to a directory and running a program. The program prompts for data for three nodes (5, 21, 3) and then displays the resulting linked list: '5 21 3 47'. The prompt 'press 1 to add next node' is shown three times, but no input is provided, leading to the final list. The terminal path is 'PS D:\Atharv\Desktop\Programming Sem-3>' and the program name is 'cd "d:\Atharv\Desktop\Programming Sem-3\DS (\$?) { .\LinkedListTraversal }'.

```
PS D:\Atharv\Desktop\Programming Sem-3> cd "d:\Atharv\Desktop\Programming Sem-3\DS ($?) { .\LinkedListTraversal }
Enter data of first node: 5

Enter data of next node: 21
press 1 to add next node: 1

Enter data of next node: 3
press 1 to add next node: 1

Enter data of next node: 47
press 1 to add next node: 0

Linked list: 5 21 3 47
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 3>
```

2. Insert at Beginning

```
#include <stdio.h>

#include <stdlib.h>

struct node
{
    int data;
    struct node *next;
};

int main()
{
    struct node *s, *p, *q;
    int ch;

    p = (struct node *)malloc(sizeof(struct node));
    printf("\nEnter data of first node :");
    scanf("%d", &p->data);

    s = p;
    do
    {
        q = (struct node *)malloc(sizeof(struct node));
        printf("\nEnter data of next node :");
        scanf("%d", &q->data);

        p->next = q;

        p = q;

        printf("\npress 1 to add next node :");
```

```

    scanf("%d", &ch);

    ;

} while (ch == 1);


p->next = NULL;

struct node *a;

a = (struct node *)malloc(sizeof(struct node));

printf("\nEnter new node data to insert at beginning :\n");

scanf("%d", &a->data);

a->next = s;

s = a;

printf("\nLinked list : ");

while (s != NULL)

{

    printf(" %d ", s->data);

    s = s->next;

}

}

```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR

PS D:\Atharv\Desktop\Programming Sem-3> cd "d:\Atharv\Desktop\Programming Sem-3\DSA\Practical 3\" ;

Enter data of first node :45

Enter data of next node :63

press 1 to add next node :1

Enter data of next node :48

press 1 to add next node :0

Enter new node data to insert at beginning :
99

Linked list : 99 45 63 48
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 3> |
```

3. Insert at End

```
#include <stdio.h>

#include <stdlib.h>

struct node
{
    int data;
    struct node *next;
};

int main()
{
    struct node *p, *q, *s, *first;
    int ch;
    p = (struct node *)malloc(sizeof(struct node));
    printf("Enter first node data :");
```

```

scanf("%d", &p->data);

s = p; // s points to the first node

first = s; // first keeps track of the start of the list

do{

    q = (struct node *)malloc(sizeof(struct node));

    printf("Enter next node data :");

    scanf("%d", &q->data);

    p->next = q;

    p = q;

    printf("\npress 1 for next node :");

    scanf("%d", &ch);

}while (ch == 1);

p->next = NULL;

// Insert a node at the end

struct node *a;

a = (struct node *)malloc(sizeof(struct node));

printf("\nEnter node data to insert at end :");

scanf("%d", &a->data);

while (s->next != NULL)

{

    s = s->next;

}

s->next = a;

a->next = NULL;

s = first;

printf("\nLinked list : ");

```

```

while (s != NULL)
{
    printf(" %d ", s->data);

    s = s->next;
}
}

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 3> cd "d:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4\"
{ .\InsertEnd }
Enter first node data :4
Enter next node data :12

press 1 for next node :1
Enter next node data :98

press 1 for next node :0

Enter node data to insert at end :56

Linked list: 4 12 98 56
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4>

```

4. Delete at Beginning

```

#include <stdio.h>

#include <stdlib.h>

struct node
{
    int data;

    struct node *next;
};

int main()
{
    struct node *s, *p, *q, *first;

    char ch;

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

```

```
printf("Enter first node data : ");
scanf("%d", &p->data);
s = p;
// s points to the first node
first = s;
// first keeps track of the start of the list
do
{
    q = (struct node *)malloc(sizeof(struct node));
    printf("Enter next node data : ");
    scanf("%d", &q->data);
    p->next = q;
    p = q;
    printf("\npress 1 for next node : ");
    scanf("%d", &ch);
} while (ch == 1);
p->next = NULL;
if (s == NULL)
{
    printf("Underflow");
}
s = first;
first = first->next;
free(s);
s = first;
printf("\nLinked list after deleting 1st node: ");
```



```

while (s != NULL)
{
    printf(" %d ", s->data);

    s = s->next;
}
}

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR

PS D:\Atharv\Desktop\Programming Sem-3> cd "d:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4\"
Enter first node data : 6
Enter next node data : 41

press 1 for next node : 1
Enter next node data : 57

press 1 for next node : 1
Enter next node data : 95

press 1 for next node : 1
Enter next node data : 32

press 1 for next node : 0

Linked list after deleting 1st node: 41 57 95 32
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4>

```

5. Delete at End

```

#include <stdio.h>

#include <stdlib.h>

struct node
{
    int data;

    struct node *next;
};

int main()
{

```

```
struct node *s, *p, *q, *first;

int ch;

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

printf("Enter first node data : ");

scanf("%d", &p->data);

s = p;

// s points to the first node

first = s;

// first keeps track of the start of the list

do
{
    q = (struct node *)malloc(sizeof(struct node));

    printf("Enter next node data : ");

    scanf("%d", &q->data);

    p->next = q;

    p = q;

    printf("\npress 1 for next node : ");

    scanf("%d", &ch);
} while (ch == 1);

p->next = NULL;

while (s->next != NULL)
{
    p = s;

    s = s->next;
}

p->next = NULL;
```

```

free(s);

s = first;

printf("\nLinked List after deleting last node: ");

while (s != NULL)
{
    printf(" %d ", s->data);

    s = s->next;
}
}

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR

PS D:\Atharv\Desktop\Programming Sem-3> cd "d:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4" ;
Enter first node data : 74
Enter next node data : 12

press 1 for next node : 1
Enter next node data : 64

press 1 for next node : 1
Enter next node data : 84

press 1 for next node : 1
Enter next node data : 24

press 1 for next node : 0

Linked List after deleting last node: 74 12 64 84
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4>

```

6. Delete at any position

```

#include<stdio.h>

#include<stdlib.h>

struct node{

int data;

struct node * next;

};

int main()

```

```

{
    struct node *s,*p,*q,*first;

    int ch;

    int key;

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

    printf("Enter first node data : ");

    scanf("%d",&p->data);

    s=p;

    // s points to the first node

    first =s ;

    // first keeps track of the start of the list

    do{

        q=(struct node *)malloc(sizeof(struct node));

        printf("Enter next node data : ");

        scanf("%d",&q->data);

        p->next=q;

        p=q;

        printf("\npress 1 for next node : ");

        scanf("%d",&ch); }while(ch==1);

        p->next=NULL;

        printf("Enter data of node to delete : ");

        scanf("%d",&key);

        while(s->data!=key)

        {

            p=s;

            s=s->next;

```

```

}

p->next=s->next;

free(s);

s=first;

printf("\nLinked list after deleting node of given data: ");

while(s!=NULL)

{

printf(" %d ",s->data);

s=s->next;

}

}

```



The screenshot shows a Windows Command Prompt window with the following text:

```

PS D:\Atharv\Desktop\Programming Sem-3> cd "d:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4\" ; if ($?) { gcc DeleteAny.c -o DeleteAny }

Enter first node data : 45
Enter next node data : 31

press 1 for next node : 1
Enter next node data : 74

press 1 for next node : 1
Enter next node data : 59

press 1 for next node : 1
Enter next node data : 33

press 1 for next node : 1
Enter next node data : 25

press 1 for next node : 0
Enter data of node to delete : 74

Linked list after deleting node of given data: 45 31 59 33 25
PS D:\Atharv\Desktop\Programming Sem-3\DSA\Practical 4>

```