DS PRACTICAL NO. :- 04

AIM: Implement a singly linked list and perform the operation like insertion, deletion and traversal.

PROGRAM:-

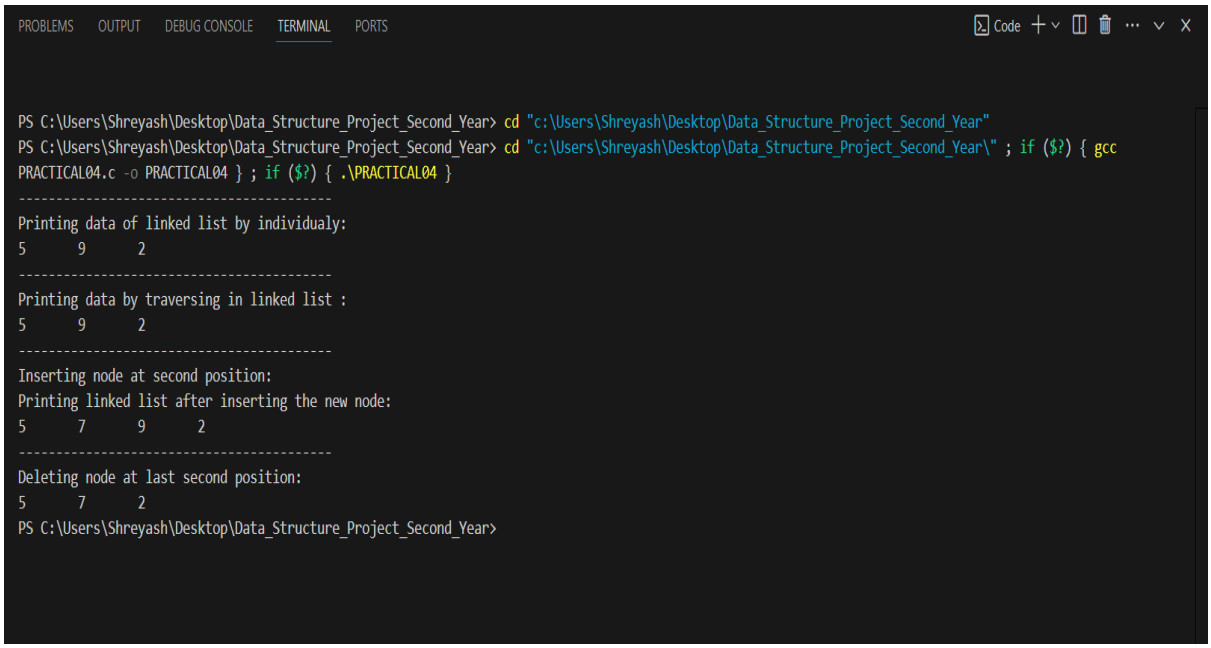
```
#include<stdio.h>
#include<stdlib.h>

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

void main(){

int index;
struct node *a;
struct node *b;
struct node *c;
struct node *d;

//Allocate memory for nodes in linked list in Heap
a = (struct node*)malloc(sizeof(struct node));
b = (struct node*)malloc(sizeof(struct node));
c = (struct node*)malloc(sizeof(struct node));
d = (struct node*)malloc(sizeof(struct node));

//Linked list
a -> data = 5;
a -> link = b;

b -> data = 9;
b -> link = c;
```

```
c -> data = 2;
c -> link = NULL;

//Printing data of linked list :-
printf("-----------------------------------------\n");
printf("Printing data of linked list by individualy:\n");
printf("%d\t",a -> data );
printf("%d\t",a -> link -> data );
printf("%d\n",b -> link -> data );

printf("-----------------------------------------\n");
//Traversing in linked list:
// Firstly getting address of first node:
struct node* p = a;

printf("Printing data by traversing in linked list :\n");
while(p != NULL){
   printf("%d\t", p -> data);
   p = p -> link;
}

//Insertion in linked list:
printf("\n");
printf("-----------------------------------------\n");
printf("Inserting node at second position:\n");
d -> data = 7;
d -> link = a -> link;
a -> link = d;

//Printing linked list after inserting the new node:
printf("Printing linked list after inserting the new node:\n");
```

```
struct node* q = a;
while(q != NULL){
   printf("%d\t", q -> data);
   q = q -> link;
}

//Deletion in linked list:
printf("\n");
printf("-----------------------------------------\n");
printf("Deleting node at last second position:\n");
struct node* k = a;

//Getting the element that has to be delete:
while(k -> data != 9){
   k = k -> link;
};

//Linking the nodes after deletion:
d -> link = b -> link;
//Printing the elements after deleting:
struct node* m = a;
while(m != NULL){
   printf("%d\t", m -> data);
   m = m -> link;
}

}
```

DS PRACTICAL NO. :- 04

OUTPUT



GITHUB LINK OF PRACTICAL NO 4 :

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