G. H. RAISONI COLLEGE OF ENGG., NAGPUR
(An Autonomous Institute under UGC Act 1956)
Department of Artificial Intelligence

Date: 05/08/2020

Practical Subject: Data Structures and Algorithms
Session: 2020-21

Student Details:

Roll Number	58
Name	Shivam Tawari
Semester	3
Section	A
Branch	Artificial Intelligence

Practical Details: Practical Number- 4

Practical Aim	Design, develop and implement a program in C that uses functions to perform the following: a) Create a singly linked list of integers. b) Delete a given integer from the above linked list. c) Display the contents of the above list after deletion.
Theory	**Linked List:** The linked list is a linear data structure that contains a sequence of elements such that each element links to its next element in the sequence. Each element in a linked list is called "Node". **Single Linked List:** Single linked list is a sequence of elements in which every element has link to its next element in the sequence. In any single linked list, the individual element is called as "Node". Every "Node" contains two fields, data field, and the next field. The data field is used to store actual value of

	the node and next field is used to store the address of next node in the sequence. The following operations are performed on a Single Linked List: • Insertion • Deletion • Display **Insertion:** In a single linked list, the insertion operation can be performed in three ways. 1. Inserting at Beginning of the list 2. Inserting at End of the list 3. Inserting at Specific location in the list **Deletion:** In a single linked list, the deletion operation can be performed in three ways. 1. Deleting from Beginning of the list 2. Deleting from End of the list 3. Deleting a Specific Node **Display:** In a single linked list, the display operation can be display the linked list present at that moment.
Procedure	1. START 2. Ask user for list size 3. Insert elements in Linked List from head 4. Display Linked List 5. Ask user for deleting an element 6. Find and delete the node 7. Display new Linked List 8. STOP

Algorithm	Step 1: START Step 2: Ask user for list size n and initialize i = 0 Step 3: Start with an empty list; point head to NULL Step 4: Enter an element var and pass it to push function Step 5: Allocate and put data in node Step 6: Make next of new node as head Step 7: Move the head to point to the new node Step 8: Increment I by 1 Step 9: While i < size of list n, go to Step 4 else go to Step 10 Step 10: Display the list Step 11: Ask user for deleting an element var Step 12: Call delete Node function with var Step 13: Find previous node of the node to be deleted Step 14: Change the next of previous node Step 15: Free memory for the node to be deleted Step 16: Display New Linked List Step 17: STOP

Program	main.c Run
	1 #include <stdio.h>
	2 #include <stdlib.h>
	3
	4 struct Node
	5 {
	6 int data;
	7 struct Node *next;
	8 };
	9
	10 void push(struct Node** head_ref, int new_data)
	11 {
	12 struct Node* new_node = (struct Node*) malloc(sizeof(struct Node
	)));
	13 new_node->data = new_data;
	14 new_node->next = (*head_ref);
	15 (*head_ref) = new_node;
	16 }
	17
	18 void deleteNode(struct Node **head_ref, int key)
	19 {
	20 struct Node* temp = *head_ref, *prev;
	21 if (temp != NULL && temp->data == key)
	22 {
	23 *head_ref = temp->next;
	24 free(temp);
	25 return;

```
    }
    while (temp != NULL && temp->data != key)
    {
        prev = temp;
        temp = temp->next;
    }
    if (temp == NULL) return;
    prev->next = temp->next;

    free(temp);
}

void printList(struct Node *node)
{
    while (node != NULL)
    {
        printf(" %d ", node->data);
        node = node->next;
    }
}

int main()
{
    struct Node* head = NULL;
    int var, n, i;
    printf("\n Name: Shivam Tawari");
    printf("\nSection: A");
    printf("\nRoll Number: 58");
    printf("\nEnter size of list: ");
    scanf("%d", &n);
    for(i=0; i<n; i++)  {
        printf(" Enter element %d: ", i+1);
        scanf("%d", &var);
        push(&head, var);
    }

    printf(" Created Linked List: ");
    printList(head);
    printf("\n Enter element to be deleted: ");
    scanf("%d", &var);
    deleteNode(&head, var);
    printf("\n Linked List after Deletion of %d: ", var);
    printList(head);
    return 0;
}
```

Output	Output Clear
	`gcc -o /tmp/m3q9M806PH.o /tmp/m3q9M806PH.c -lm`
	`/tmp/m3q9M806PH.o`
	`Name: Shivam Tawari`
	`Section: A`
	`Roll Number: 58`
	`Enter size of list: 7`
	`Enter element 1: 5`
	`Enter element 2: 9`
	`Enter element 3: 88`
	`Enter element 4: 2`
	`Enter element 5: 65`
	`Enter element 6: 24`
	`Enter element 7: 58`
	`Created Linked List: 58 24 65 2 88 9 5`
	`Enter element to be deleted: 24`
	`Linked List after Deletion of 24: 58 65 2 88 9 5`

Conclusion	Hence, successfully designed and developed a program to create a Linked List and performed basic operations – insertion, delete and display on it.