**DS Algorithm Essentials | December 2020**

**Assignment Day 3 | 26th December 2020**

**Name : P. Visurthi**

**Question 1**

**Write a function “insert\_any()” for inserting a node at any given position of the linked list. Assume position starts at 0.**

**Question 2**

**Write a function “delete\_beg()” for deleting a node from the beginning of the linked list.**

**Question 3**

**Write a function “delete\_end()” for deleting a node from the end of the linked list.**

**Code :**

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**#include <stdio.h>**

**#include<stdlib.h>**

**#include<string.h>**

**int count = 0;**

**struct node**

**{**

**char name[20],branch[20],usn[10];**

**int sem;**

**char phno[10];**

**struct node \*next;**

**}\*first = NULL, \*last = NULL, \*temp = NULL;**

**void create ()**

**{**

**char name[20], usn[10],branch[20];**

**int sem;**

**char phno[10];**

**temp = (struct node \*) malloc (sizeof (struct node));**

**printf ("Enter the student details \n");**

**printf ("\nName, USN, Branch,Sem,Phone Number :");**

**scanf ("%s%s%s%d%s", temp->name, temp->usn, temp->branch,&temp->sem,temp->phno);**

**temp->next = NULL;**

**count++;**

**}**

**insert\_any(struct node \*a, int value)**

**{**

**char name[20], usn[10],branch[20];**

**int sem;**

**char phno[10];**

**temp = (struct node \*) malloc (sizeof (struct node));**

**printf ("Enter the student details \n");**

**printf ("\nName, USN, Branch,Sem,Phone Number :");**

**scanf ("%s%s%s%d%s", temp->name, temp->usn, temp->branch,&temp->sem,temp->phno);**

**temp->next = NULL;**

**count++;**

**if(first == NULL)**

**{**

**first = temp;**

**last = first;**

**}else**

**{**

**temp -> next = first;**

**first = temp;**

**}**

**}**

**void delete\_Beg()**

**{**

**temp=first;**

**if(first==NULL)**

**{**

**printf("\n list is empty");**

**return;**

**}**

**if(temp->next==NULL)**

**{**

**printf("The deleted node is \n");**

**printf("%s\t%s\t%s\t%d\t%s",temp->name,temp->usn,temp->branch,temp->sem,temp->phno);**

**free(temp);**

**first=NULL;**

**}else**

**{**

**first=temp->next;**

**printf("The deleted node is \n");**

**printf("%s\t%s\t%s\t%d\t%s",temp->name,temp->usn,temp->branch,temp->sem,temp->phno);**

**free(temp);**

**}**

**count--;**

**}**

**void delete\_End()**

**{**

**temp=first;**

**if(first==NULL)**

**{**

**printf("\n list is empty");**

**return;**

**}**

**if(temp->next==NULL)**

**{**

**printf("The deleted node is \n");**

**printf("%s\t%s\t%s\t%d\t%s",temp->name,temp->usn,temp->branch,temp->sem,temp->phno);**

**free(temp);**

**first=NULL;**

**}else**

**{**

**while(temp->next!=last)**

**temp=temp->next;**

**printf("The deleted node is \n");**

**printf("%s\t%s\t%s\t%d\t%s",last->name,last->usn,last->branch,last->sem,last->phno);**

**free(last);**

**last=temp;**

**last->next=NULL;**

**}**

**count--;**

**}**

**void main()**

**{**

**int ch,i,n;**

**while(1)**

**{**

**printf("\n\n1.Insert n details student\n2.Delete from beginning\n3.Delete from last");**

**printf("\n Enter your choice : ");**

**scanf("%d",&ch);**

**switch(ch)**

**{**

**case 1 : printf("\nEnter the value of n ");**

**scanf("%d",&n);**

**for(i=0;i<n;i++)**

**insert\_any(n,i);**

**break;**

**case 2 : delete\_Beg();**

**break;**

**case 3 : delete\_End();**

**break;**

**default: printf("\n Wrong Input, try again");**

**}**

**}**

**}**