**Name: Rohit Ghosh.**

**Roll Number: 21051420.**

**Date: 08/08/2022.**

**Class Assignment**

**Program 2**

**Question:**

WAP to display the m’th Node from the Last of the Single Linked List of n Nodes.

**Code:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <stdint.h>

#include <stdbool.h>

//all packages are imported now

typedef struct node

{

int data;

struct node \*next;

}\*NODE;

//we made the structure

NODE create(NODE start)

{

//this method creates a new node

NODE new\_node;

int data;

printf("\nEnter the data. If entered -1, list is complete \n");

scanf("%d",&data);

while(data!=-1)

{

new\_node=(struct node\*)malloc(sizeof(struct node\*));

new\_node->data=data;

if(start==NULL)

{

//this is for empty list

start=new\_node;

new\_node->next=NULL;

}

else //making it the first node

{

//here the nodes are inserted from the first

new\_node->next=start;

start=new\_node;

}

printf("\nEnter the data. If entered -1, list is complete \n");

scanf("%d",&data);

}

return start;

}

//end of creating node method

int count\_node(NODE start)

{

//here the number of nodes are counted

NODE ptr=start;

int count=0;

while(ptr!=NULL)

{

ptr=ptr->next;

count++;

}

return count;

}

//end of count\_node method

void search(NODE start,int pos)

{

//this method searches an item entered by the user

int count=count\_node(start);

NODE ptr=start;

int i=1;//loop variable

int flag=0;//flag variable to check the variable found or not

while(ptr!=NULL)

{

if(i==pos)

{

flag=1;

break;

//flag is updated to 1 for getting the item in the list

}

else

{

ptr=ptr->next;

i++;

}

}

if(flag==0)

{

printf("\nData Not Fount\n");

}

else

{

printf("\nThe Data: %d\n",ptr->data);

}

}

//end of search method

int main()

{

//this is the main execution method which controls the flow of execution

NODE start=NULL;

//main execution function

int data;

start=create(start);

printf("\nEnter the position to be searched\n");

int pos;

scanf("%d",&pos);

search(start,pos);

return 0;

}

//end of main method

**Output:**

