QUESTION 1 – Write A Function "insert\_any()" For Inserting A Node At Any Given Position Of The Linked List.

Ans –

#include<stdio.h>

typedef struct node

{

int data;

struct node \*next;

}node;

struct node \*head=NULL;

void insert(int data)

{

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

temp->data=data;

temp->next=head;

head=temp;

}

void insert\_any(int data,int position)

{

node \*ptr = (node\*)malloc(sizeof(node));

ptr->data=data;

int i;

node \*temp=head;

if(position==1)

{

ptr->next=temp;

head=ptr;

return;

}

for(i=1;i<position-1;i++)

{

temp=temp->next;

}

ptr->next=temp->next;

temp->next=ptr;

}

void print()

{

node \*temp=head;

while(temp!=NULL)

{

printf("\n%d ",temp->data);

temp=temp->next;

}

}

void main()

{

int pos,dd;

printf("\n \nEntered List Is - ");

insert(1);

insert(2);

insert(3);

print();

printf("\n \nEnter The Position -");

scanf("%d",&pos);

printf("\n \nEnter Data -");

scanf("%d",&dd);

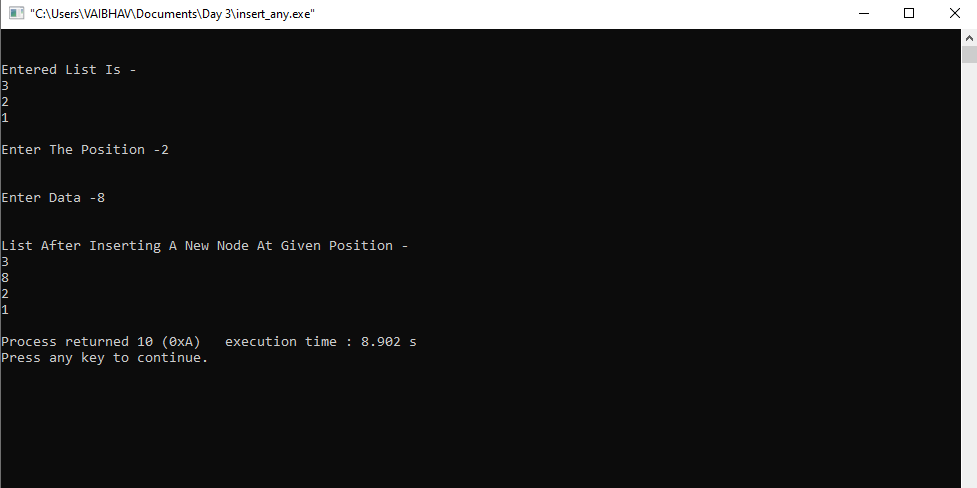
insert\_any(dd,pos);

printf("\n \nList After Inserting A New Node At Given Position -");

print();

printf("\n"); }

Output –



Question 2 – Write A Function "delete\_beg()" For Deleting A Node From The Begining Of The Linked List.

Ans –   
 #include<stdio.h>

typedef struct node

{

int data;

struct node \*next;

}node;

struct node \*head=NULL;

void delete\_beg()

{

node\* temp=head;

node\* holder;

if(temp!=NULL)

{

holder=temp->next;

free(temp);

head=holder;

}

}

void print()

{

node \*temp=head;

while(temp!=NULL)

{

printf("\n%d ",temp->data);

temp=temp->next;

}

}

void main()

{

node \*one = (node\*)malloc(sizeof(node));

node \*two = (node\*)malloc(sizeof(node));

node \*three = (node\*)malloc(sizeof(node));

head=one;

printf("\n \nEntered List is - ");

one->data=1;

one->next=two;

two->data=2;

two->next=three;

three->data=3;

three->next=NULL;

print();

printf("\n \nList After Deleting A Node From The Beginging - ");

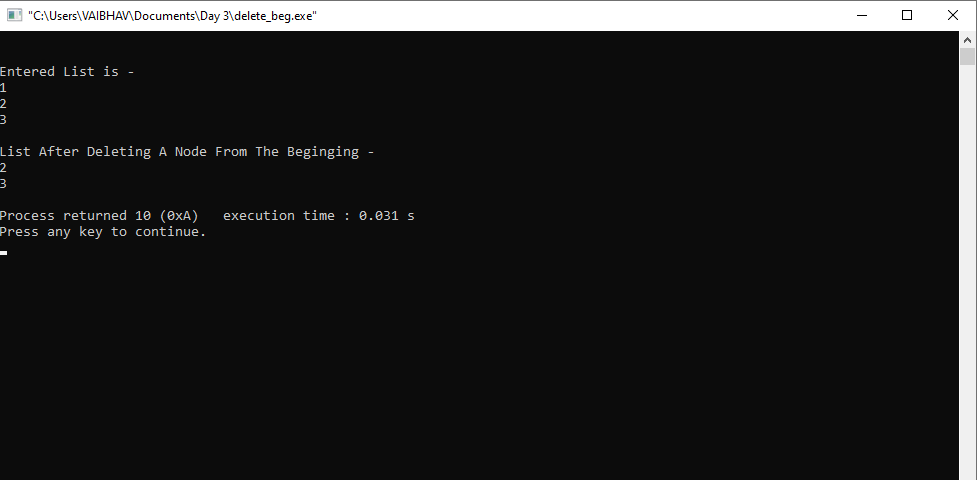
delete\_beg();

print();

printf("\n");

}

Output –



QUESTION 3 – Write A Function "delete\_end()" For Deleting A Node From The End Of The Linked List.

Ans –

#include<stdio.h>

typedef struct node

{

int data;

struct node \*next;

}node;

struct node \*head=NULL;

void delete\_end()

{

node \*temp=head;

node \*holder=head;

if(temp!=NULL)

{

temp=temp->next;

if(temp==NULL)

{

head=NULL;

free(holder);

return;

}

while(temp!=NULL)

{

if(temp->next!=NULL)

{

temp=temp->next;

holder=holder->next;

}

else

{

break;

}

}

holder->next=NULL;

free(temp);

}

}

void print()

{

node \*temp=head;

while(temp!=NULL)

{

printf("\n%d ",temp->data);

temp=temp->next;

}

}

void main()

{

node \*one = (node\*)malloc(sizeof(node));

node \*two = (node\*)malloc(sizeof(node));

node \*three = (node\*)malloc(sizeof(node));

head=one;

one->data=1;

one->next=two;

two->data=2;

two->next=three;

three->data=3;

three->next=NULL;

printf("\n \nEntered List Is - ");

print();

printf("\n \nList After Deleting The Node From The End - ");

delete\_end();

print();

printf("\n");

}

Output –

