// Write a C program to create a double linked list by inserting nodes such that linked remains in ascending order (Using double pointer).

/\*

NAME : Kuldeep Rao

COURSE : MCA(4TH Sem)

ROLL NO. : 1101660

\*/

#include<stdio.h>

#include<stdlib.h>

typedef struct node{

struct node \*pvs;

int data;

struct node \*next;

}nodetype;

void insert(nodetype\*\*,nodetype\*\*); //Function Declaration

void display(nodetype\*);

int main(){

nodetype \*head=NULL,\*tail=NULL;

int opt =1;

while(opt<3&&opt>0){

printf("\nEnter your choice \n 1 INSERT \n 2 DISPLAY \n PRESS ANY KEY TO EXIT\n");

scanf("%d",&opt);

if(opt==1)

insert(&head,&tail); //Function calling

if(opt==2)

display(head);

}

return 0;

}

void insert(nodetype \*\*hd,nodetype \*\*tl){

int num;

nodetype \*p=NULL,\*temp=NULL;

printf("\nEnter the element to be pushed :");

scanf("%d",&num);

p=(nodetype\*)malloc(sizeof(nodetype));

p->data=num;

if((\*hd)==NULL){ //Inserting first node

\*hd=\*tl=p;

(\*hd)->pvs=NULL;

(\*hd)->next=NULL;

}

else{

temp=\*hd;

while(temp!=NULL){

/\* Comparing the new nodes after the first node is inserted \*/

if(num < temp->data){

/\* If the incoming node’s data is the smallest \*/

if(temp == \*hd)

\*hd = p;

else

(temp->pvs)->next=p;

p->pvs=temp->pvs;

p->next=temp;

temp->pvs=p;

break;

}

temp=temp->next;

}

/\* If the incoming node’s data is the greatest of all the nodes present \*/

if(temp==NULL){

(\*tl)->next=p;

p->pvs=(\*tl);

p->next=NULL;

(\*tl)=p;

}

}

}

void display(nodetype \*head){

if(head==NULL)

printf("\nstack is empty");

else{

while(head!=NULL){

printf("%d\t",head->data);

head=head->next;

}

}

}