#include<stdio.h>

#include<stdlib.h>

#include<string.h>

struct node

{

int salary;

char a[20];

int id;

struct node \*next;

struct node \*prev;

}\*root=NULL,\*newnode,\*temp1,\*temp2;

void insert(struct node \*root,struct node \*newnode)

{

if(newnode->salary>root->salary)

{

if(root->next==NULL)

root->next=newnode;

else

insert(root->next,newnode);

}

else if(newnode->salary<root->salary)

{

if(root->prev==NULL)

root->prev=newnode;

else

insert(root->prev,newnode);

}

printf("Node added successfully in BS tree\n");

printf("\_\_\_\_\_\_\_\_\_\_\n");

}

int total(struct node \* temp1)

{

int sum=0;

if(temp1!=NULL)

{

sum=(temp1->salary)+total(temp1->prev)+total(temp1->next);;

return sum;

}

}

void max\_min()

{ temp1=temp2=root;

while(temp1!=NULL && temp1->next!=NULL)

{

temp1=temp1->next;

}

while(temp2!=NULL && temp2->prev!=NULL)

{

temp2=temp2->prev;

}

printf("maximun salary:%d\nminimun salary:%d\n",temp1->salary,temp2->salary);

printf("\_\_\_\_\_\_\_\_\_\_\n");

}

void display(struct node \* temp1)

{

if(temp1!=NULL)

{

display(temp1->prev);

printf("\n%s\n",temp1->a);

printf("%d\n",temp1->id);

printf("%d\n",temp1->salary);

display(temp1->next);

}

}

void display\_range(struct node \* temp1)

{

int n1,n2;

printf("ENTER SALARY RANGE1:");

scanf("%d",&n1);

printf("ENTER SALARY RANGE2:");

scanf("%d",&n2);

if(temp1!=NULL)

{ display(temp1->prev);

if(temp1->salary>=n1 && temp1->salary<=n2)

{

printf("%d",temp1->salary);

}

display(temp1->next);

}

}

int main()

{

int c,sum;

while(1)

{

printf("1.insert\n2.max\_min\n3.total monthly expenses\n4.Display the names of employee based on salary\n5.Display the names of employee based on salary at specific range\n6.exit");

printf("\nenter the choice:");

scanf("%d",&c);

switch(c)

{

case 1:

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

newnode->next=NULL;

newnode->prev=NULL;

printf("enter the name:");

scanf("%s",&newnode->a);

printf("enter id:");

scanf("%d",&newnode->id);

printf("enter salary:");

scanf("%d",&newnode->salary);

if(root==NULL)

root=newnode;

else

insert(root,newnode);

break;

case 2:

max\_min();

break;

case 3:

sum=total(root);

printf("%d\n",sum);

break;

case 4:display(root);

break;

case 5:display\_range(root);

break;

case 6:exit(0);

break;

}

}

}