PROGRAM 10

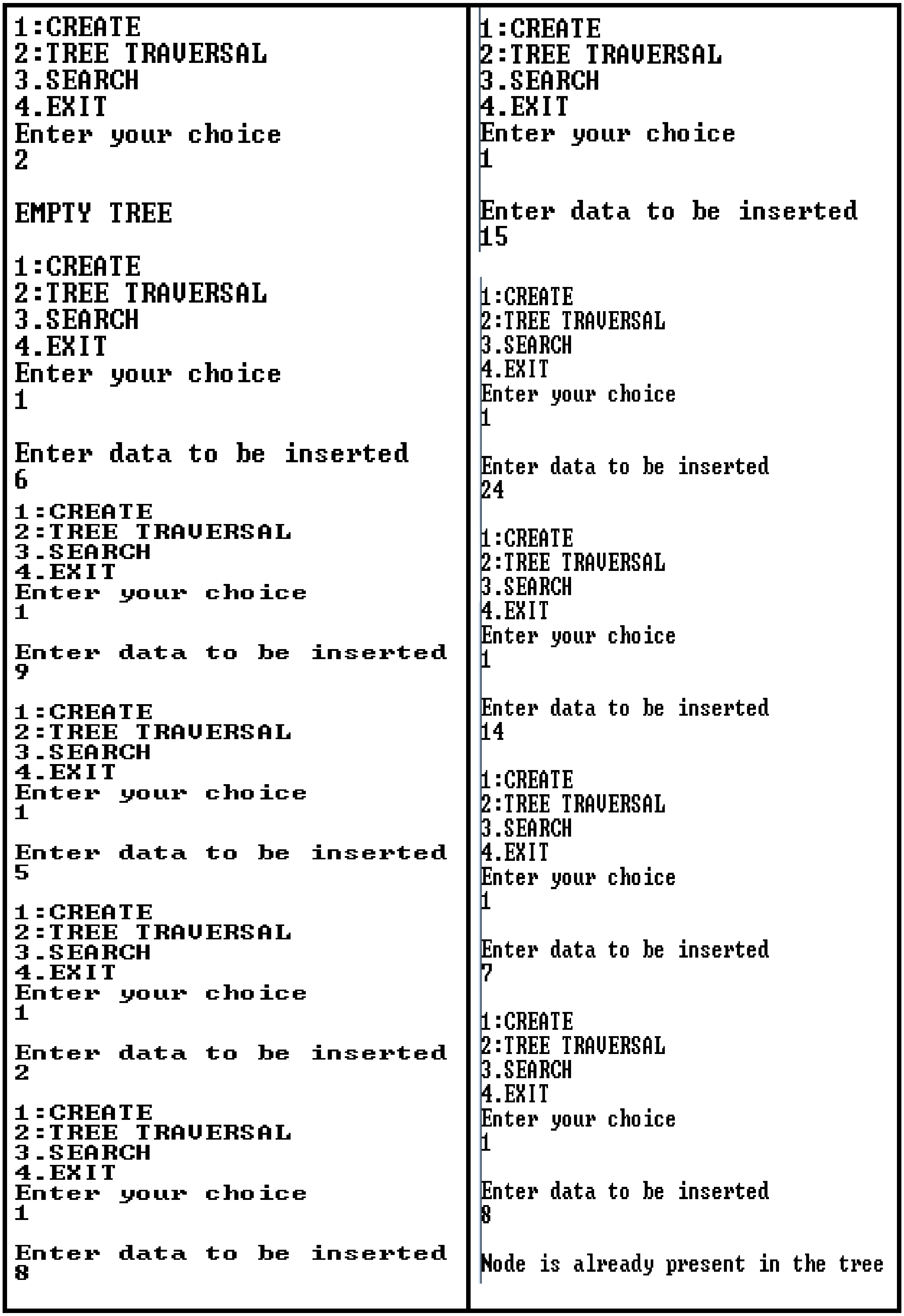
Design, Develop and Implement a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers

- **a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2**
- **b. Traverse the BST in In-order, Preorder and Post Order**
- **c. Search the BST for a given element (KEY) and report the appropriate message**
- **d. Exit**

```
#include<stdio.h>
#include<stdlib.h>
int choice,data,key;
struct node
{
int info;
struct node *lchild,*rchild;
};
typedef struct node *NODE;

int main()
{
NODE root=NULL;
NODE CREATE(NODE,int);
void INORDER(NODE),POSTORDER(NODE),PREORDER(NODE);
NODE SEARCH_NODE(NODE,int);

while(1)
{
printf("\n1:CREATE\n2:TREE TRAVERSAL\n3.SEARCH\n4.EXIT");
printf("\nEnter your choice\n");
scanf("%d",&choice);
switch(choice)
{
case 1: printf("\nEnter data to be inserted\n");
scanf("%d",&data);
root=CREATE(root,data);
break;

case 2: if(root==NULL)
printf("\nEMPTY TREE\n");
else
{
printf("\nThe Inorder display : ");
INORDER(root);
printf("\nThe Preorder display : ");
PREORDER(root);
printf("\nThe Postorder display : ");
POSTORDER(root);
}
break;
case 3: printf("\nenter the key to search:\n");
scanf("%d",&key);
SEARCH_NODE(root,key);
break;

case 4:  exit(0);
}
}
}

NODE  CREATE(NODE root,int data)
{
NODE newnode,x,parent;
newnode=(NODE)malloc(sizeof(struct node));
newnode->lchild=newnode->rchild=NULL;
newnode->info=data;

if(root==NULL)
root=newnode;
else
{
x=root;
while(x!=NULL)
{
parent=x;
if(x->info<data)
x=x->rchild;
else if(x->info>data)
x=x->lchild;
else
{
printf("\nNode is already present in the tree\n");
return(root);
}
}
if(parent->info<data)
parent->rchild=newnode;
else
parent->lchild=newnode;
}
return(root);
}

void INORDER(NODE root)
{
if(root!=NULL)
{
INORDER(root->lchild);
```

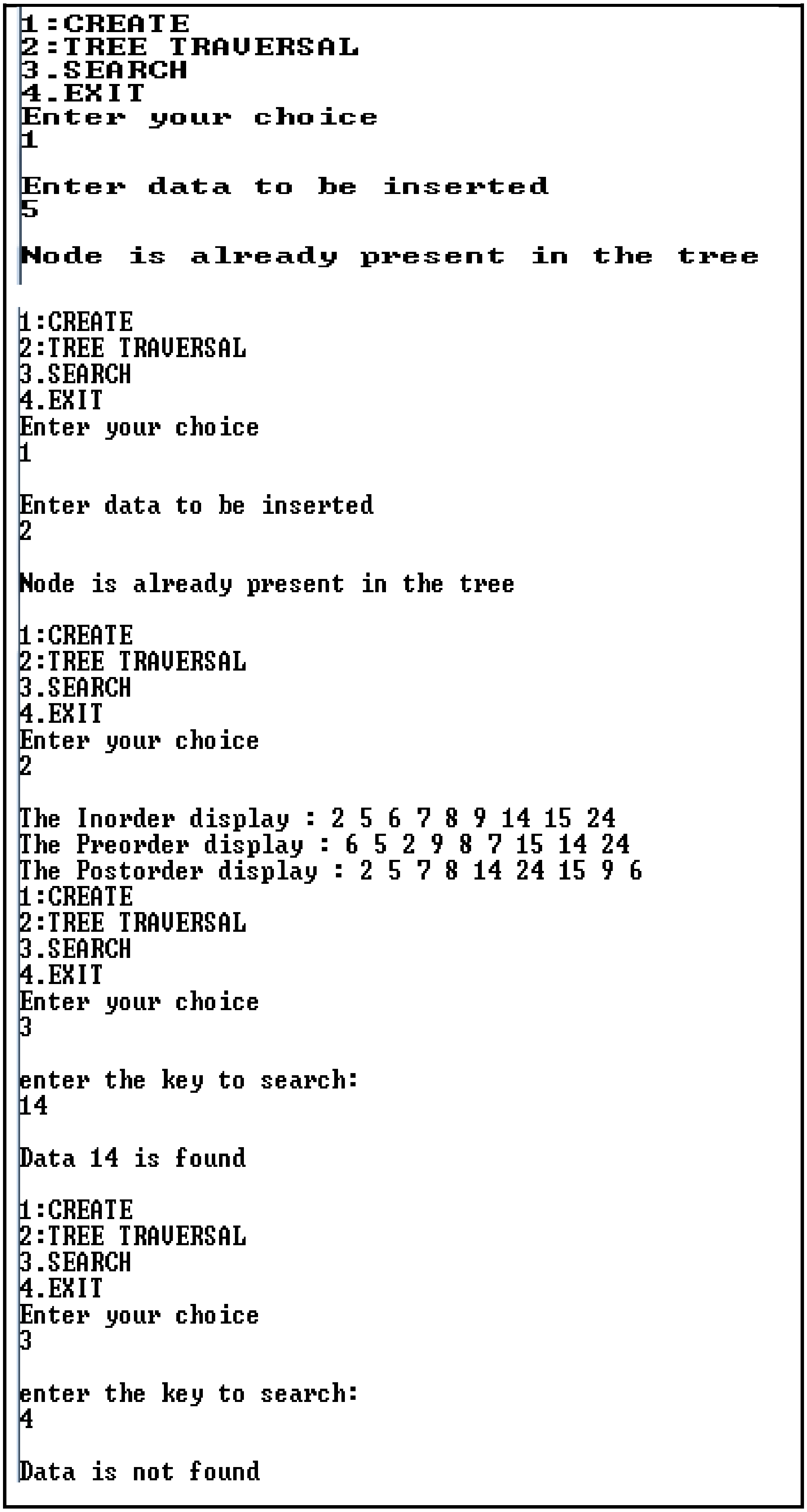
```
printf("%d ",root->info);
INORDER(root->rchild);
}
}

void PREORDER(NODE root)
{
if(root!=NULL)
        {
printf("%d ",root->info);
PREORDER(root->lchild);
PREORDER(root->rchild);
        }
}

void POSTORDER(NODE root)
{
if(root!=NULL)
{
POSTORDER(root->lchild);
POSTORDER(root->rchild);
printf("%d ",root->info);
}
}

NODE SEARCH_NODE(NODE root, int key)
{
NODE cur,q,parent,successor;
if(root==NULL)
{
printf("\nTree is empty\n");
return root;
}
parent=NULL,cur=root;
while(cur!=NULL)
{
if(key==cur->info)
break;
parent=cur;
cur= (key<cur->info)?cur->lchild:cur->rchild;
}
if(cur==NULL)
{
printf("\nData is not found\n");
return root;
}
printf("\nData %d is found\n",key);
}
```

OUTPUT:



```
1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
2

EMPTY TREE

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
6
1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
9

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
5

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
2

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
8
```

```
1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
15

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
24

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
14

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
7

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
8

Node is already present in the tree
```



```
1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
5

Node is already present in the tree

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
1

Enter data to be inserted
2

Node is already present in the tree

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
2

The Inorder display : 2 5 6 7 8 9 14 15 24
The Preorder display : 6 5 2 9 8 7 15 14 24
The Postorder display : 2 5 7 8 14 24 15 9 6
1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
3

enter the key to search:
14

Data 14 is found

1:CREATE
2:TREE TRAVERSAL
3.SEARCH
4.EXIT
Enter your choice
3

enter the key to search:
4

Data is not found
```