Question 1:

```
#include<stdio.h>
#include<stdlib.h>
#include <string.h>

typedef struct node
{
   char srn[13];
   struct node*llink;
   struct node*rlink;

}node_t;

typedef struct tree
{
   node_t* root;
}tree_t;

void init(tree_t*pt)
{
   pt->root=NULL;

}

void create(tree_t*pt,int n)
{

   node_t*temp;
   node_t*pres;
   node_t*prev;
  printf("Enter root node: ");
```

```
    pt->root=(node_t*)malloc(sizeof(node_t));
    scanf("%s",(pt->root->srn));
    pt->root->llink=pt->root->rlink=NULL;

    do
    {
        printf("Enter node value: ");
        temp=(node_t*)malloc(sizeof(node_t));
        scanf("%s",(temp->srn));
        temp->llink=temp->rlink=NULL;
        prev=NULL;
        pres=pt->root;

        while(pres!=NULL)
        {
            prev=pres;
            if(strcmp(temp->srn,pres->srn)<0)
                pres=pres->llink;
            else
                pres=pres->rlink;
        }
        if(strcmp(temp->srn,prev->srn)<0)
            prev->llink=temp;
        else
            prev->rlink=temp;
         --n;
    } while(n>0);

}

void inorder_traversal(node_t*p)
```

```
{
   if(p!=NULL)
   {


      inorder_traversal(p->llink);
      printf("%s\n",p->srn);
      inorder_traversal(p->rlink);
   }
}


 int search(node_t*p,char* ele)
{
   int found=0;
   if(p!=NULL)
   {


      search(p->llink,ele);
      if(strcmp(p->srn,ele)==0)
      {


         found=1;
         return(found);
      }
      search(p->rlink,ele);
   }
   return(found);

}
```

```
int main()
{
   tree_t t;
   init(&t);
   printf("Enter number of nodes: ");
   int len;
   scanf("%d",&len);
   create(&t,len-1);
   inorder_traversal(t.root);
   printf("\n Enter srn to search for: ");
   char ele[13];
   scanf("%s",ele);
   int res=search(t.root,ele);
   if(res==1)
      printf("\n %s is found \t",ele);
   else
      printf("\n Node not found");

  return 0;

}
```

```
Enter number of nodes: 3
Enter root node: PES1UG20CS500
Enter node value: PES1UG20CS510
Enter node value: PES1UG20CS106
PES1UG20CS106
PES1UG20CS500
PES1UG20CS510

 Enter srn to search for: PES1UG20CS100

 Node not found
Press any key to continue . . .
```

Question 2:

```
#include<stdio.h>
typedef struct tree_node
{
int info;
int used;
}TREE;
#define MAXNODES 50
void init(TREE t[MAXNODES])
{
for(int i=0;i<MAXNODES;i++)
t[i].used=0;
}
int create(TREE *bst)
{
int ele, wish;
printf("Enter the root element\n");
scanf("%d",&bst[0].info);
bst[0].used=1;
int cnt=1;
do{
printf("Enter an element\n");
scanf("%d",&ele);
int p=0;

while(p<MAXNODES && bst[p].used)
{
if(ele<bst[p].info)
p=2*p+1;
else
p=2*p+2;
```

```
}
if(p>=MAXNODES)
printf("Insertion not possible\n");
else
{
bst[p].info=ele;
bst[p].used=1;
cnt++;
}
printf("Do you wish to add another\n");
scanf("%d",&wish);
}while(wish);
return cnt/2;
}
void preorder(TREE* bst, int r)
{
if(bst[r].used)
{
printf("%d ",bst[r].info);
preorder(bst,2*r+1);
preorder(bst,2*r+2);
}
}
int main()
{
TREE bst[MAXNODES];
init(bst);
int height=create(bst);
printf("The height of the tree is %d and the level is %d\n",height+1,height);
preorder(bst,0);
return 0;
```

}

```
Enter the root element
45
Enter an element
12
Do you wish to add another
1
Enter an element
96
Do you wish to add another
1
Enter an element
75
Do you wish to add another
1
Enter an element
23
Do you wish to add another
0
The height of the tree is 3 and the level is 2
45 12 23 96 75
```