

```
void print_root()
{
    int i,j;
    cout<<"各子树的根: "<<endl;
    for(i=1;i<=N;i++)
    {
        for(j=1;j<=N;j++)
            cout<<root[i][j]<<" ";
        cout<<endl;
    }
}

void construct_optimal_bst(int i,int j)
{
    if(i<=j)
    {
        int r=root[i][j];
        cout<<r<<" ";
        construct_optimal_bst(i,r-1);
        construct_optimal_bst(r+1,j);
    }
}
```

```

void print_bst(int i,int j)
{
    if(i==1&&j==N)
        cout<<"root is "<<root[i][j]<<endl;
    if(i<j)
    {
        int r=root[i][j];
        if(i!=r)
            cout<<"left child root "<<root[i][r-1]<<endl;
        print_bst(i,root[i][j]-1);
        if(j!=r)
            cout<<"right child root "<<root[r+1][j]<<endl;
        print_bst(root[i][j]+1,j);
    }
}

int main()
{
    optimal_bst_search_tree(p,q,N);
    print_root();
    cout<<"构造的最优二叉树: "<<endl;
    construct_optimal_bst(1,5);
    cout<<endl;
    print_bst(1,5);
}

```