```
#include
<stdio.h>
            #include <limits.h>
            int sum(int freq[], int i, int j);
            int optCost(int freq[], int i, int j)
            {
            if (j < i)
             return 0;
            if (j == i)
             return freq[i];
            int fsum = sum(freq, i, j);
            int min = INT_MAX;
            for (int r = i; r <= j; ++r)
            {
             int cost = optCost(freq, i, r-1) +
                optCost(freq, r+1, j);
             if (cost < min)
              min = cost;
            }
            return min + fsum;
            }
            int optimalSearchTree(int keys[], int freq[], int n)
            {
             return optCost(freq, 0, n-1);
            }
            int sum(int freq[], int i, int j)
            {
             int s = 0;
             for (int k = i; k <=j; k++)
             s += freq[k];
             return s;
            }
            int main()
            {
                int n,L1[20],L2[20],i;
                printf("enter the size of elements :");
                scanf("%d",&n);
                printf("enter key:");
                for(i=0;i<n;i++)
                scanf("%d",&L1[i]);
              printf("enter Frequency:");
                for(i=0;i<n;i++)
                scanf("%d",&L2[i]);
             printf("Cost of Optimal BST is %d ",
               optimalSearchTree(L1, L2, n));
             return 0;
```

```
}
```