

Pract 6  
A8\_b2\_20

```
#include <stdio.h>
#include <float.h>
#define INFINITY FLT_MAX
void optimal_BST(float p[],float q[],int n,float e[][n+1],int root[][n+1])
{
    float w[n+2][n+1];
    int i,j,l,r;
    float t;
    for(i=1;i<=n+1;i++)
    {
        e[i][i-1]=q[i-1];
        w[i][i-1]=q[i-1];
    }
    for(l=1;l<=n;l++)
    {
        for(i=1;i<=n-l+1;i++)
        {
            j=i+l-1;
            e[i][j]=INFINITY;
            w[i][j]=w[i][j-1]+p[j]+q[j];
            for(r=i;r<=j;r++)
            {
                t=e[i][r-1]+e[r+1][j]+w[i][j];
                if(t<e[i][j])
                {
                    e[i][j]=t;
                    root[i][j]=r;
                }
            }
        }
    }
}

int main() {
    int n = 4;
    float p[]={1,0.1,0.2,0.4,0.3};
    float q[]={0.05,0.1,0.05,0.05,0.1};
    float e[n+2][n+1];
    int root[n+1][n+1];
    optimal_BST(p,q,n,e,root);
    printf("%f\n",e[1][n]);
    printf("root matrix\n");
    for(int i=1;i<=n;i++)
    {
        for(int j=i;j<=n;j++)
```

```

    {
        printf("%d ",root[i][j]);
    }
    printf("\n");
}
return 0;
}

```

main.c

```

42     int n = 4;
43     float p[]={1,0.1,0.2,0.4,0.3};
44     float q[]={0.05,0.1,0.05,0.05,0.1};
45     float e[n+2][n+1];
46     int root[n+1][n+1];
47     optimal_BST(p,q,n,e,root);
48     printf("%f\n",e[1][n]);
49     printf("root matrix\n");
50     for(int i=1;i<=n;i++)
51     {
52         for(int j=i;j<=n;j++)
53         {
54             printf("%d ",root[i][j]);
55         }
56         printf("\n");
57     }
58     return 0;
59 }

```

2.900000

root matrix

1 2 2 3

2 3 3

3 3

4

...Program finished with exit code 0

Press ENTER to exit console.