#include<iostream>

using namespace std;

#define MAX 10

int find(int i,int j);

void print(int,int);

int w[10][10],c[10][10],r[10][10],i,j,k,n,m;

float p[MAX],q[MAX];

char idnt[MAX][10];

int main()

{

cout <<"\nenter the no of identifiers : ";

cin>>n;

cout <<"\nenter identifiers";

for(i=1;i<=n;i++)

{

cout<<"\nIdentifier no "<<i<<" : ";

cin>>idnt[i];

}

cout <<"\nenter success propability for identifiers";

for(i=1;i<=n;i++)

{

cout<<"\np["<<i<<"] : ";

cin>>p[i];

}

cout << "\nenter failure propability for identifiers";

for(i=0;i<=n;i++)

{

cout<<"\nq["<<i<<"] : ";

cin >> q[i];

}

for(i=0;i<=n;i++)

{

w[i][i]=q[i];

c[i][i]=r[i][i]=0;

w[i][i+1]=q[i]+q[i+1]+p[i+1];

r[i][i+1]=i+1;

c[i][i+1]=q[i]+q[i+1]+p[i+1];

}

w[n][n]=q[n];

r[n][n]=c[n][n]=0;

for(m=2;m<=n;m++)

{

for(i=0;i<=n-m;i++)

{

j=i+m;

w[i][j]=w[i][j-1]+p[j]+q[j];

k=find(i,j);

r[i][j]=k;

c[i][j]=w[i][j]+c[i][k-1]+c[k][j];

}

}

cout<<"\nRoot of this OBST : "<<r[0][n]<<"\tthat is\t"<<idnt[r[0][n]];

cout<<"\nCost of this OBST : "<<c[0][n];

cout<<"\nIn preorder ";

cout <<"\n";

print(0,n);

return 0;

}

int find(int i,int j)

{

int min=2000,m,l;

for(m=i+1;m<=j;m++)

if(c[i][m-1]+c[m][j]<min)

{

min=c[i][m-1]+c[m][j];

l=m;

}

return l;

}

void print(int i,int j)

{

if(i<j)

cout<<"\n"<<idnt[r[i][j]];

else

return;

print(i,r[i][j]-1);

print(r[i][j],j);

}

**OUTPUT:**

enter the no of identifiers : 4

enter identifiers

Identifier no 1 : if

Identifier no 2 : while

Identifier no 3 : for

Identifier no 4 : do

enter success propability for identifiers

p[1] : 0.7

p[2] : 0.5

p[3] : 0.1

p[4] : 0.2

enter failure propability for identifiers

q[0] : 0.3

q[1] : 0.5

q[2] : 0.9

q[3] : 0.8

q[4] : 0.4

Root of this OBST : 1 that is if

Cost of this OBST : 5

In preorder

if

while

for

do