**矩阵乘法**

问题描述

　　有n个矩阵，大小分别为a0\*a1, a1\*a2, a2\*a3, ..., a[n-1]\*a[n]，现要将它们依次相乘，只能使用结合率，求最少需要多少次运算。  
　　两个大小分别为p\*q和q\*r的矩阵相乘时的运算次数计为p\*q\*r。

输入格式

　　输入的第一行包含一个整数n，表示矩阵的个数。  
　　第二行包含n+1个数，表示给定的矩阵。

输出格式

　　输出一个整数，表示最少的运算次数。

样例输入

3  
1 10 5 20

样例输出

150

数据规模和约定

　　1<=n<=1000, 1<=ai<=10000。

**C++代码：**

#include <iostream>

#include <cstring>

#include <cmath>

using namespace std;

typedef long long ll;

ll d[1010][1010];

struct m{

ll x,y;

}p[1010];

ll n,a[1010];

ll dp(int n){

memset(d,0,sizeof(d));

for(int len=1;len<n;len++){

for(int i=0,j=i+len;j<n;j++,i++){

ll min=9999999999999999;

for(int k=i;k<j;k++){

ll count =d[i][k]+d[k+1][j]+p[i].x\*p[k].y\*p[j].y;

min=(count<min)?count:min;

}

d[i][j]=min;

}

}

return d[0][n-1];

}

int main(int argc, char\*\* argv) {

cin>>n;

for(int i=0;i<=n;i++)scanf("%lld",&a[i]);

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

p[i].x=a[i];

p[i].y=a[i+1];

}

printf("%lld\n",dp(n));

return 0;

}