1 Classical Problems

区间染色 II 代码实现。

```
# include <cstdlib>
# include <cstdio>
# include <cmath>
# include <cstring>

int n=0;

struct person_t
{
        float p_;
        person_t(float pp=0):p_(pp)
        {

        }
        float p (void)
        {
                return p_;
        };
        float q(void)
        {
                return 1-p_;
        };
};


person_t merge_r(person_t x,person_t y)
{
        return person_t(1-y.q()/(1-x.q()*y.p()));
};
```

```
person_t merge_l(person_t x,person_t y)
{
        return person_t(x.p()/(1-x.q()*y.p()));
};

person_t a[100005],b[100005] ,c[100005];

int main(void)
{
        scanf("%d",&n);
        for(int i=1;i<=n;++i) scanf("%lf",&a[i].p_);
        if(n==3)
        {
            //something 只看第一个人向左边传还是右边传
        }
        b[2]=a[2];
        for(int i=3;i<=n;++i) b[i]=merge_l(b[i-1],a[i]);
        c[n]=a[n];
        for(int i=n-1;i>=1;--i) c[i]=merge_r(a[i],c[i+1]);
        //ANSWER i=1;
        //ANSWER i=2;
        for(int i=3;i<=n-1;++i)
        {
                if(i+2<=n) p =merge_r(c[i+2],p);
                if(i-2>=2) p =merge_l(p,b[i-2]);
                float ans=0;
                ans+=p.p()*a[i+1].q(),p.q()/(1-p.q()*a[i+1].q());
                ans+=p.q()*a[i-1].p(),p.q()/(1-p.q()*a[i-1].p());
        }
        return 0;
}
```