

游戏的买

[E-游戏的买 牛客小白月赛67\(nowcoder.com\)](#)

solve

根本无法理解这个问题背景：当前情况下，最优的选择是什么？概率的应用背景是什么？

主要的问题，最优的策略是什么？

其实问题可以转变成，考虑当前开始选择。然后每一个基本事件中，选出最小的一项。进行期望统计。

定义状态： dp_i 表示只考虑后*i*个的情况。

状态转移方程

$$dp_i = \min(dp_{i+1}, a_i)/2.0 + \min(dp_i, b_i)/2.0 \quad (1)$$

code

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  typedef long long ll;
4
5  const int oo = 0x0fffffff;
6  const int N = 1E6 + 10;
7
8  double x[N] , y[N];
9  double f[N];
10
11 void work(int testNo)
12 {
13     int n;
14     cin >> n;
15
16     for (int i = 1; i <= n; i++) {
17         cin >> x[i];
18     }
19
20     for (int i = 1; i <= n; i++) {
21         cin >> y[i];
22     }
23     f[n] = (x[n] + y[n]) / 2.0;
24     for (int i = n - 1; i > 0; -- i) {
25         f[i] = min(f[i + 1] , x[i]) * 0.5 + min(f[i + 1] , y[i]) * 0.5;
26     }
27     cout << fixed << setprecision(6) << f[1] << '\n';
28
29 }
30
31
32
33 int main()
```

```
34 {
35     ios::sync_with_stdio(false);
36     cin.tie(0);
37
38     int t; cin >> t;
39     for (int i = 1; i <= t; i++)work(i);
40 }
41
42 /* stuff you should look for
43 * int overflow, array bounds
44 * special cases (n=1?)
45 * do smth instead of nothing and stay organized
46 * WRITE STUFF DOWN
47 * DON'T GET STUCK ON ONE APPROACH
48 */
49
```

很抽象还是不大懂。