

ЗАДАЧА C2

Задачу можно решить «динамическим программированием».

Пусть $dp[a][b] = prob$ - текущее состояние.

a - кол-во выпавших орлов.

b - кол-во выпавших орлов.

$prob$ - вероятность попасть в это состояние.

База динамики: $dp[0][0] = 1$

Переходы динамики:

$dp[a+1][b] += dp[a][b] * p$

$dp[a][b+1] += dp[a][b] * p$

Если у состояния $a \geq 2$ и $b \geq 1$, то к ответу прибавляется $(a + b) * prob$ этого состояния.

Так как требуется точность 4 зн. после запятой, можно считать результат только для довольно небольших $(a + b)$.

Код на C++

```
#include <bits/stdc++.h>
using namespace std;

double p = 1.0 / 6;
double q = 5.0 / 6;

int main(){

    double res = 0;
    map<pair<int,int>, double> dp;
    dp[pair<int,int>(0,0)] = 1;

    for (int i = 0; i < 10000; ++i){
        if (i % 500 == 0) cerr << i << " " << res << endl;
        map<pair<int,int>, double> dpswap;
        for (auto& it : dp){
            pair<int,int> nxt;
            double next_p;

            nxt = it.first;
            nxt.first += 1;
            next_p = it.second * p;
            if (nxt.first >= 2 && nxt.second >= 1)
                res += (nxt.first + nxt.second) * next_p;
            else
                dpswap[nxt] += next_p;

            nxt = it.first;
```

```

        nxt.second += 1;
        next_p = it.second * q;
        if (nxt.first >= 2 && nxt.second >= 1)
            res += (nxt.first + nxt.second) * next_p;
        else
            dpswap[nxt] += next_p;
    }
    swap(dp, dpswap);
}

cout << res;

return 0;
}

```