Problem Link:

<https://leetcode.com/problems/soup-servings/?envType=daily-question&envId=2025-08-08>

Solution:

class Solution {

private:

vector<vector<double>> dp;

double helper(int a, int b) {

if(a <= 0 && b <= 0)

return 0.5;

if(a <= 0)

return 1.0;

if(b <= 0)

return 0.0;

if(dp[a][b] >= 0)

return dp[a][b];

dp[a][b] = 0.25 \* (

helper(max(0, a - 4), b) +

helper(max(0, a - 3), max(0, b - 1)) +

helper(max(0, a - 2), max(0, b - 2)) +

helper(max(0, a - 1), max(0, b - 3))

);

return dp[a][b];

}

public:

double soupServings(int n) {

if(n >= 5000)

return 1.0;

int N = (n + 24) / 25;

dp = vector<vector<double>>(N + 1, vector<double>(N + 1, -1.0));

return helper(N, N);

}

};