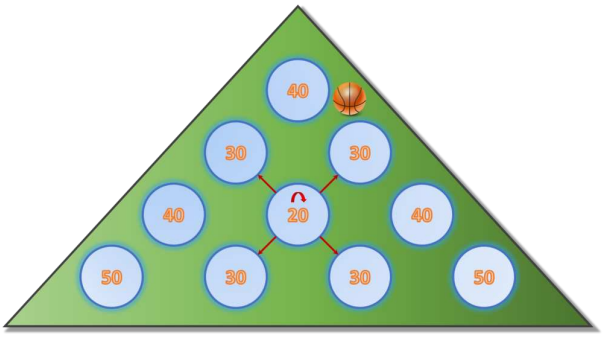
Problem B
Arcade!

Have you recently visited an arcade? Arcade games seem to have become more boring over the years, requiring less and less skill. In fact, most arcade games these days seem to depend entirely on luck. Consider the arcade game shown in the picture, which consists of different holes arranged in a triangular shape. A ball is dropped near the hole at the top. The ball either falls into the hole, in which case the game ends, or it bounces to one of its (up to) 4 neighbors, denoted by the red arrows. Different holes have different payouts — some may even be negative! If the ball reaches another hole, the process repeats: the ball either falls into the hole, ending the game — or it bounces to one of its neighbors, possibly ad infinitum!



Write a program that computes the expected payout when dropping a ball into the machine!

Input

The input consists of a single test case. The first line contains an integer N ($1 \leq N \leq 32$) describing the number of rows of the arcade machine. The second line contains $H = N(N+1)/2$ integers v_i ($-100 \leq v_i \leq 100$) describing the payout (positive or negative) if the ball drops into hole i. Holes are numbered such that hole 1 is in the first row, holes 2 and 3 are in the second row, etc. The k^{th} row starts with hole number $k(k-1)/2+1$ and contains exactly k holes.

These two lines are followed by H lines, each of which contains 5 real numbers $p_0\ p_1\ p_2\ p_3\ p_4$, denoting the probability that the ball bounces to its top-left (p_0), top-right (p_1), bottom-left (p_2), or bottom-right (p_3) neighbors or that the ball enters the hole (p_4). Each probability is given with at most 3 decimal digits after the period. It is guaranteed that $0.0 \leq p_i \leq 1.0$ and $\sum p_i = 1.0$. If a hole does not have certain neighbors because it is located near the boundary of the arcade machine, the probability of bouncing to these non-existent neighbors is always zero. For instance, for hole number 1, the probabilities to jump to the top-left and top-right neighbors are both given as 0.0.

You can assume that after the ball has bounced b times, the probability that it has not fallen into a hole is at most $(1-10^{-3})^{\lfloor b/H \rfloor}$.

Output

Output a single number, the expected value from playing one game. Your answer is considered correct if its absolute or relative error is less than 10^{-4}.

Hint: Using Monte Carlo-style simulation (throwing many balls in the machine and simulating which hole they fall into using randomly generated choices) does not yield the required accuracy!



Sample Input 1	Sample Output 1
4 40 30 30 40 20 40 50 30 30 50 0.0 0.0 0.45 0.45 0.1 0.0 0.3 0.3 0.3 0.1 0.3 0.0 0.3 0.3 0.1 0.0 0.3 0.3 0.3 0.1 0.2 0.2 0.2 0.2 0.2 0.3 0.0 0.3 0.3 0.1 0.0 0.8 0.0 0.0 0.2 0.4 0.4 0.0 0.0 0.2 0.4 0.4 0.0 0.0 0.2 0.8 0.0 0.0 0.0 0.2	32.6405451448

Sample Input 2	Sample Output 2
2 100 50 50 0.0 0.0 0.45 0.45 0.1 0.0 0.90 0.0 0.0 0.10 0.90 0.0 0.0 0.0 0.10	76.31578947368