Problem J. Longest Shortest Path

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

You are given a directed graph and two nodes s and t. The given graph may contain multiple edges between the same node pair but not self loops. Each edge e has its initial length d_e and the cost c_e. You can extend an edge by paying a cost. Formally, it costs $x \cdot c_e$ to change the length of an edge e from d_e to $d_e + x$. (Note that x can be a non-integer). Edges cannot be shortened.

Your task is to maximize the length of the shortest path from node s to node t by lengthening some edges within cost P. You can assume that there is at least one path from s to t.

Input

The first line of the input contains five integers N, M, P, s, and t: N ($2 \leq N \leq 200$) and M ($1 \leq M \leq 2,000$) are the number of the nodes and the edges of the given graph respectively, P ($0 \leq P \leq 10^6$) is the cost limit that you can pay, and s and t ($1 \leq s, t \leq N$, $s \neq t$) are the start and the end node of objective path respectively. Each of the following M lines contains four integers v_i, u_i, d_i and c_i, which mean there is an edge from v_i to u_i ($1 \leq v_i, u_i \leq N$, $v_i \neq u_i$) with the initial length d_i ($1 \leq d_i \leq 10$) and the cost c_i ($1 \leq c_i \leq 10$).

Output

Output the maximum length of the shortest path from node s to node t by lengthening some edges within cost P. The output can contain an absolute or a relative error no more than 10^{-6}.

Examples

standard input	standard output
3 2 3 1 3 1 2 2 1 2 3 1 2	6
3 3 2 1 3 1 2 1 1 2 3 1 1 1 3 1 1	2.5000000
3 4 5 1 3 1 2 1 2 2 3 1 1 1 3 3 2 1 3 4 1	4.25