First let's find every vertex's distance from 1 and call it $dis1_v$ and find also from n and call it $disn_v$ (here we can use dijkstra).
Second we construct a directed graph D with n vertices with following method:
For any edge like e that's between v and u with weight w (consider $dis1_v < dis1_u$) we draw an edge from v to u if:

- $dis1_v + w = disn_u$
- $dis1_v + w + disn_u = dis1_n \; (or \; disn_1)$

Now any path from 1 to n in D is a shortest path between 1 and n in problem's graph and vice versa.
Also D is a dag because in every D's edge $dis1$ increases (or $disn$ decreases) so there will be no cycles in D.
Now the problem becomes, we have a dag D which its only source is 1 and its only sink is n, for each vertex like v find if v is in either all paths from sink to source, or some of them or none of them.
Note that we ignore vertices with no edges connected to them in D and they are in no shortest path.

Because there is only one source and sink and all vertices degree is at least 1 in D, all not ignored vertices are at least in one path from sink to source.
Now consider topological sort of D to be $a_1, a_2, ..., a_n$ (obviously $a_1 = 1$ and $a_n = n$), for each i $(2 \leq i \leq n-1)$, a_i is in all paths if there is no edge from vertices $a_1, ..., a_{i-1}$ to $a_{i+1}, ..., a_n$ (otherwise we can jump from a_i using that edge in our path from a_1 to a_n).
For finding these vertices, consider a initially all zero array b of size n and for any edge from v to u (that $a_j = v$ and $a_k = u$) do following operations:

- $b_{j+1} = b_{j+1} + 1$
- $b_k = b_k - 1$

Now a_i is in all paths iff $\sum_{j=1}^{i} b_j$ is zero which can be calculated for all is using a simple partial sum.
In summary first we do a dijkstra in $O((n+m) \times log(n+m))$ and then some graph building and topological sort in $O(n{+}m)$ and finally a partial sum in $O(n)$.

So all can be done in $O((n+m) \times log(n+m))$.