The fastest route | 2 point(s)

*To use the words of the late Mitch Hedberg: “an escalator can never break, it can only become stairs.”*

You are at the 0th floor of a poorly designed building and need to get to a given floor for an interview. You need to find the path that will allow you to get to your interview in the least amount of time possible. You have in your possession a list of all escalators in this building and the direction in which they are running. At any given time, you can choose to stop any escalator, turn it into stairs, and walk up or down these stairs freely.

Using an escalator has a fixed cost in terms of time. The cost of walking up stairs is equal to the number of stories you are walking up multiplied by your walking speed (S \* abs(start\_floor – end\_floor)).

Clarifications

* You can assume that the building contains an infinite number of floors.
* If two paths took the same time, we prefer the path that uses the fewest escalators.
* If two escalators were equivalent, we prefer the escalator that has the smallest ID.
* If stopping or not stopping an escalator would result in the exact same time consumption, we prefer **not** stopping it.

Input description/format

One line containing the following values (all integers separated by a single white space):

* T: The floor at which your interview is taking place (your target floor) (> 0)
* E: The time it takes to take one escalator (≥ 0)
* S: The speed at which you can walk up/down stairs (≥ 0)
* N: The number of escalators in the building (≥ 0)

Followed by n lines of the following format, describing the **directed** escalators of the building (meaning you can only move from the start floor to the end floor and not vice versa, unless you stop the escalator) (all integers ≥ 0, all separated by a single white space):

* The ID of the escalator (a sequential number that goes from 0 to N-1)
* The starting floor of the escalator
* The end floor of the escalator (different from the starting floor)

Output description/format

If there is no path to get to your interview, output the string Invalid.

If there is a solution, print the following three lines:

1. The string STOPPED: followed by a space and (white space separated) escalator IDs that are meant to be stopped (**sorted** from smallest to largest), or “None” if no escalator was stopped.
2. The string TIME: followed by a space and the time it will take you to get to the interview.
3. The string PATH: followed by (white space separated) escalator ID’s in the order that they are visited.

Example input

55 5 1 5

0 0 5

1 5 56

2 5 56

3 57 55

4 56 57

Example output

STOPPED: 3 4

TIME: 13

PATH: 0 1 4 3