CCC '20 S2 - Escape Room (Hard)

Time limit: 3.0s **Memory limit:** 512M

You have to determine if it is possible to escape from a room. The room is an M-by-N grid with each position (cell) containing a positive integer. The rows are numbered $1, 2, \ldots, M$ and the columns are numbered $1, 2, \ldots, N$. We use (r, c) to refer to the cell in row r and column c.

You start in the top-left corner at $(1, 1)$ and exit from the bottom-right corner at (M, N). If you are in a cell containing the value x, then you can jump to any cell (a, b) satisfying $a \times b = x$. For example, if you are in a cell containing a 6, you can jump to cell $(2, 3)$.

Note that from a cell containing a 6, there are up to four cells you can jump to: $(2, 3)$, $(3, 2)$, $(1, 6)$, or $(6, 1)$. If the room is a 5-by-6 grid, there isn't a row 6 so only the first three jumps would be possible.

Note: The constraints and data have changed from the original problem.

Implementation details

You should implement the following procedure:

```
bool can_escape(int M, int N, std::vector<std::vector<int>> v)
```

- M: the number of rows in the room.
- N: the number of columns in the room.
- v: a two-dimensional array of integers representing the values of the cells.
- This procedure should return true if it is possible to escape, and false otherwise.

Example

Consider the following call.

```
can_escape(3, 4, {{0, 0, 0, 0, 0},
                  {0, 3, 10, 8, 1},
                  {0, 1, 11, 12, 12},
                  {0, 6, 2, 3, 9}})
```

Starting in the cell at $(1, 1)$ which contains a 3, one possibility is to jump to the cell at $(1, 3)$. This cell contains an 8 so from it, you could jump to the cell at $(2, 4)$. This brings you to a cell containing 12 from which you can jump to the exit at $(3, 4)$. Note that another way to escape is to jump from the starting cell to the cell at $(3, 1)$ to the cell at $(2, 3)$ to the exit.

This call should return true.

Constraints

For all subtasks:

$2 \le M, N \le 4\,000$

v is an $M + 1$ by $N + 1$ array.

The first row and column of v consists of 0's. The remaining integers are between 1 and $M \times N$, inclusive.

Subtask 1 [20%]

$M, N \le 1\,000$

Subtask 2 [30%]

$M, N \le 2\,000$

Subtask 3 [30%]

$M, N \le 3\,000$

Subtask 4 [20%]

No additional constraints.