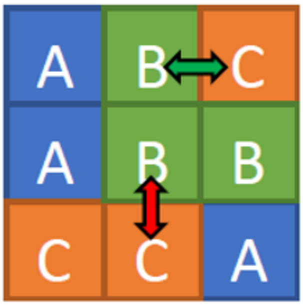
Work

The work is inspired by a land permutation problem. The lands are squares and have equal value. Lands that are neighbors can be exchanged. Owners intend to make land swaps to reduce the number of borders. They do not accept any swaps that increase the number of borders. If there is a swap that results in a greater reduction of the number of borders, they do not accept another one with a smaller reduction.

The problem can be described as follows: an instance is an N x M matrix of K-colored squares. The matrix represents the map, and the colors represent the owners, identified by letters or numbers. Two squares of different colors sharing a side create a border. The goal is to reduce the number of borders. An action is a land swap, which consists of exchanging the colors of two contiguous squares. This action is valid only if the number of borders does not increase, and if there is no other action that leads to a greater reduction of the number of borders. The objective is to find a sequence of actions that leads to a number of borders equal to or less than a value W.

Example instance: 3x3 matrix of 3 colors. The goal is to find a solution with 6 or fewer borders.



The current state has 4 vertical borders and 4 horizontal borders, totaling 8 borders. Each border can result in an action of swapping the colors of the two squares, but this action is only valid if the number of borders does not increase as a result of the swap, and if there is no other action that leads to a greater reduction.

There are two border actions marked. The red action would result in an increase in the number of borders. Square B had 2 borders, and in the new position, it would have 3, while square C had 2 borders and would have 4. Since the number of borders increases, the action is not valid.

The green action has square C with 2 borders and square B with 2 borders. When they are swapped, square C in the new position has 3 borders, while square B has only 1 border. Thus, the number of borders remains at 4, making it a valid action, as there is currently no other action that reduces the number of borders.

Consider the following parameters for 7 instances:

ID	N	M	K	W_1	W_2
1	3	3	3	6	5
2	2	4	2	4	2
3	4	4	3	10	9
4	4	5	2	10	9
5	4	10	4	30	25
6	8	10	2	41	35
7	8	10	8	70	62

Each instance has two versions, version 1 (W1) and version 2 (W2). [Challenge: for students who have a working implementation, and after completing their report, they can try to solve the last 3 instances with W values of 22, 21, and 56, respectively, considering this a third version.]

In addition to the parameters, the initial coloring of the squares is defined for each instance, with each color identified by an integer number, with the letter A associated with the number 1, and so on.

ID	Mapa	ID	Mapa
1	1 2 3 1 2 2 3 3 1 Fronteiras: 8	3	1 2 2 2 1 3 3 3 1 2 1 1 1 1 3 2 Fronteiras: 15
2	1 2 2 2 1 2 1 1 Fronteiras: 5		
4	1 1 2 1 1 2 2 1 2 1 1 1 2 1 2 2 1 1 2 1 Fronteiras: 24	5	1 2 2 2 2 1 2 2 2 2 1 3 3 3 4 1 3 3 3 4 1 2 1 4 3 1 2 1 4 3 1 4 4 4 3 1 4 4 4 3 Fronteiras: 42
6	1 1 2 1 1 1 1 2 1 1 2 2 1 2 1 2 2 1 2 1 1 1 2 1 2 1 1 2 1 2 2 1 1 2 1 2 1 1 2 1 1 1 2 1 1 1 1 2 1 1 2 2 1 2 1 2 2 1 2 1 1 1 2 1 2 1 1 2 1 2 2 1 1 2 1 2 1 1 2 1 Fronteiras: 108	7	1 1 2 8 8 1 4 3 1 4 2 2 1 8 3 8 4 3 2 1 1 1 8 8 3 1 6 2 1 4 2 1 1 3 1 2 1 1 4 4 1 7 7 3 1 1 5 6 4 4 2 2 1 3 1 2 2 1 6 6 1 7 2 7 5 5 5 5 1 6 2 7 7 7 1 5 5 1 6 6 Fronteiras: 102

The instance maps can be defined as matrices with a maximum value of 10x10 and can be statically initialized in the code.

```
{
{{1,2,3},{1,2,2},{3,3,1}},
{{1,2,2,2},{1,2,1,1}},
{{1,2,2,2},{1,3,3,3},{1,2,1,1},{1,1,3,2}},
{{1,1,2,1,1},{2,2,1,2,1},{1,1,2,1,2},{2,1,1,2,1}},
```

```
{{1,2,2,2,2,1,2,2,2,2},{1,3,3,3,4,1,3,3,3,4},{1,2,1,4,3,1,2,1,4,3},{1,
4,4,4,3,1,4,4,4,3}},

{{1,1,2,1,1,1,1,2,1,1},{2,2,1,2,1,2,2,1,2,1},{1,1,2,1,2,1,1,2,1,2},{2,
1,1,2,1,2,1,1,2,1},{1,1,2,1,1,1,1,2,1,1},{2,2,1,2,1,2,2,1,2,1},{1,1,2,
1,2,1,1,2,1,2},{2,1,1,2,1,2,1,1,2,1}},

{{1,1,2,8,8,1,4,3,1,4},{2,2,1,8,3,8,4,3,2,1},{1,1,8,8,3,1,6,2,1,4},{2,
1,1,3,1,2,1,1,4,4},{1,7,7,3,1,1,5,6,4,4},{2,2,1,3,1,2,2,1,6,6},{1,7,2,
7,5,5,5,5,1,6},{2,7,7,7,1,5,5,1,6,6}}
}
```

Instance ID=1 is the example instance. To achieve 6 borders, a set of valid actions must be taken. One possible solution can be reported as follows (if there are more than 15 actions, only show the initial and final states):

A B B

A A B

C C C

Boarders: 6 (1)

A B B

A B A

C C C

Boarders: 7 (2)

A B B

A B C

C C A

Boarders: 8 (3)

A B C

A B B

C C A

Boarders: 8

Blind searches must be used to solve the e-portfolio, which means that successors cannot be ordered, and no other type of heuristic information can be used. Of course, successors that are proven to not lead to a solution can be excluded.

You should deliver:

- Report;
- Source code of the implemented algorithms.

The report should contain a table with the results of the execution of the tested algorithms/configurations vs the provided instances. For each algorithm/instance, it should show:

- Number of expansions;
- Number of generations;
- Result - impossible; solution; not solved.
- Time spent (not exceeding 1 minute).

For each instance, in the result of each execution, one of three situations may occur: obtaining proof that the instance is impossible; solving the instance and presenting the solution in that case; not solving and exceeding the time limit. In the end, you should have the best information obtained considering all the executions, meaning that you present the best result obtained for each instance.