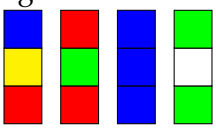
Carpets

Your great aunt Maude has, over the years, acquired a large collection of strips of carpet squares. She wants to start a business of sewing them together into larger carpets for sale. Of course people's tastes differ so the criteria that make a carpet desirable can also change depending on the customer or type of carpet. You'll be trying to satisfy customer demand as well as you possibly can from the stock of material available.

In any given scenario, you will have a supply of some number of strips of differently coloured squares, all of the same length. For instance you might have:



You will make carpets by sewing these strips together along their long sides. The criteria you need to satisfy will vary. It might be strictly forbidden to have two squares of the same colour sewn together. Or, it might be desirable to have as many such coincidences as possible or the best possible balance between coincidences and differences.

The trouble is that you don't know in advance what a customer will ask for, so you need to leave dear Maude with a working program into which she can enter:

- her current stock of carpet pieces,
- the carpet size requested, and
- the criteria for judging a carpet's aesthetics.

The program will then tell her which items of stock to use and how to fit them together. Detailed requirements are included in the task description below.

Task

Write a program `makeCarpet` which takes as input from `stdin` a stock of carpet pieces represented as lines of characters, each line of the same length. Use a command line argument to give the size of carpet desired (a positive integer equal to the number of pieces to be used) and a command line flag to represent the type. You must support the following options:

-n No matches allowed.
-m Maximum possible number of matches.
-b Best balance between matches and non-matches.

In each case output should be a carpet of the required type (or "Not possible" in the first case) followed by a line which contains just the number of matching squares.

(3 points, Group)