# J · PSET

Time Limit: 5 seconds
Memory Limit: 128MB

PSET is a derivative of the game *SET*. The game *SET* has 81 cards, each of which has one, two or three of the same shapes. The shapes are (for this problem):

Arrow: **D**iamond: and **O**val: .

Each group of shapes will have a color Red, Green or Blue (labeled **R**, **G** or **B** in case this page is black and white) and a fill type:

Empty: **S**triped: or **F**illed: .

This gives 81 possible combinations. Three cards are a *SET*, if, for each property (count, color, fill and shape), the property is the same on all three cards or different on all three cards. For example, for the following cards, **1**, **2** and **3** form a *SET* (different counts, same color, different fill, different shape) but **1**, **2** and **4** do not form a *SET* (for several reasons, one of which is **1** and **4** have the same count, **2** has a different count):

| 1 | 2 | 3 | 4 |
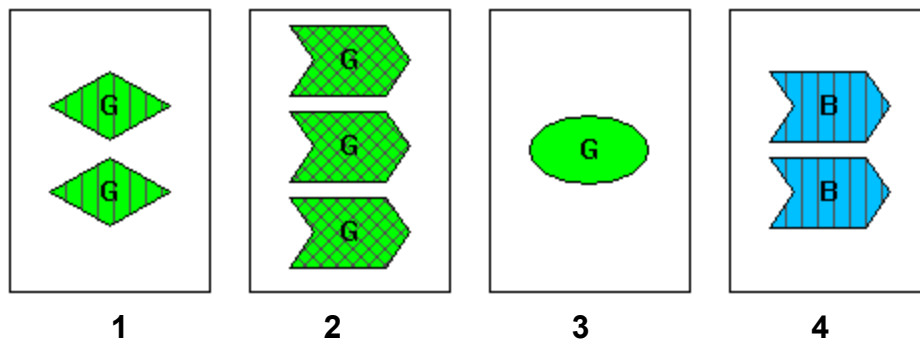
Note that given two cards, there is exactly one other card which forms a set with the first two.

We will use the code `{count}{color}{fill}{shape}` to specify a *SET* card. For example, the cards above are: `2GSD`, `3GFA`, `1GEO`, `2BSA`. {fill} is one of E, S or F for Empty, Striped or Filled respectively. {shape} is one of A, D or O for Arrow, Diamond or Oval respectively.

Each *PSET* card consists of two set cards different from `2GSD` which form a *SET* with `2GSD`. From the example above `3GFA` and `1GEO`. The *SET* cards on the *PSET* card are above one another and rotated 90 degrees. See the example below.

Three **PSET** cards form a **PSET** if the (possibly after flipping a card) the top *SET* cards form a *SET* and the bottom *SET* cards form a *SET*.

In the example below, there are four **PSET**s. { **1**, (flip) **2,3** }, { **1**, **2**, **4**}, { (flip) **1**, **3**, **4**} and { **2**, **3** , (flip)**4**.

1   2   3   4
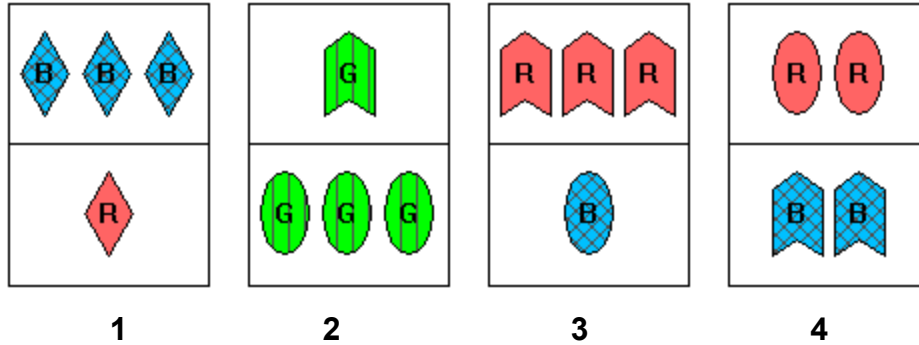
Write a program which takes as input a collection of distinct **PSET** cards and outputs the number of (three card) **PSET**s.

Input

Input consists of multiple lines of input. The first line contains the number **N** of **PSET** cards to follow (**4 ≤ N ≤ 20**). This is followed by **N** lines of input, one per card. Each card line consists of a four character code (as described above) for the top of the card followed by a space and a four character code for the bottom of the card.

Output

The output consists of a single line that contains the integer number of **PSET**s in the input collection.

Sample 1:

Sample Input	Sample Output
4 3BFD 1RED 1GSA 3GSO 3REA 1BFO 2REO 2BFA	4