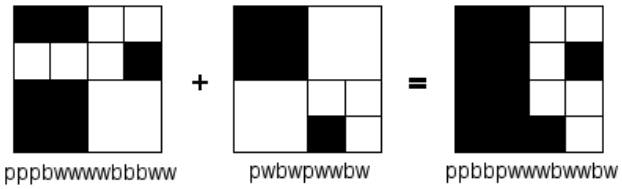
Challenge 11 - The escape from Pixel Island

As a NASA engineer, assigned to the Landsat Data Continuity Mission, you have been informed that the satellite images from the Scabb Island present weird patterns and possible something hidden in them. The following information was sent to your department yesterday:

> Hi dear fellow,
>
> In my research about salmon fishing in the Yemen and Caribbean Islands I recently found something quite remarkable in the supposedly uninhabited Scabb Island. I can see moving pixels from the satellite images received every 12 hours. I attach a drawing of the first one I found.

> I managed to create a program that reads the satellite images and convert them into a format that you may be able to analyze with your resources and expertise.
>
> My intuition tells me that you need to add up the weird black squares and look for a pattern. Good Luck!

As the letter said, you need to write a program that adds up the squares and finds the hidden message.



Input

First line contains the number of test cases, **T**, and **T** cases follow (each one in a different line). Each test case consists of several strings separated by whitespaces. Each string contains only the characters **p**, **w** and **b** (see image above) and represents one pattern of squares.

Output

For each test case, output the secret message (on a different line).

Submit & test your code

To test and submit code we provide a set of tools to help you. Download contest tools if you haven't already done that. You will then be able to test and submit your solution to this challenge with the challenge token.

```
Challenge token: _nbMozhBSaQ1va8fQrwS
```

To test your program

```
./test_challenge _nbMozhBSaQ1va8fQrwS path/program
```

A nice output will tell you if your program got the right solution or not. You can try as many times as you need.

To submit your program to the challenge

```
./submit_challenge _nbMozhBSaQ1va8fQrwS path/source_pkg.tgz path/program
```

Note that you first need to solve the test phase before submitting the code. During the submit phase, in some problems, we might give your program harder questions, so try to make your program failsafe.

Important: In this phase, you must provide the source code used to solve the challenge and, if necessary, a brief explanation of how you solved it.

Remember **you can only submit once!** Once your solution is submitted you won't be able to amend it to fix issues or make it faster, so please be sure your solution is finished before submitting it.

If you have any doubts, please check the info section.

Go ahead

I'm done! :)

Once you have submitted your code, hit refresh and continue to next challenge.

I'm stuck! :(

Be sure you follow the Tuenti Engineering twitter for updates and possible hints during the contest.

If this challenge is too hard and you are blocked, you will be able to skip it after two hours. Note that **you won't be able to complete it later**, and you have a limited number of challenges to skip.

Finally, if you run out of skips but are still really stuck with one problem, you will be able to skip it after 24 hours.

Challenge status:

Test case	Not done
Solution submitted	Not done
Skip	You still have to wait 0h, 30m and 0s to be able to skip this challenge

Refresh status