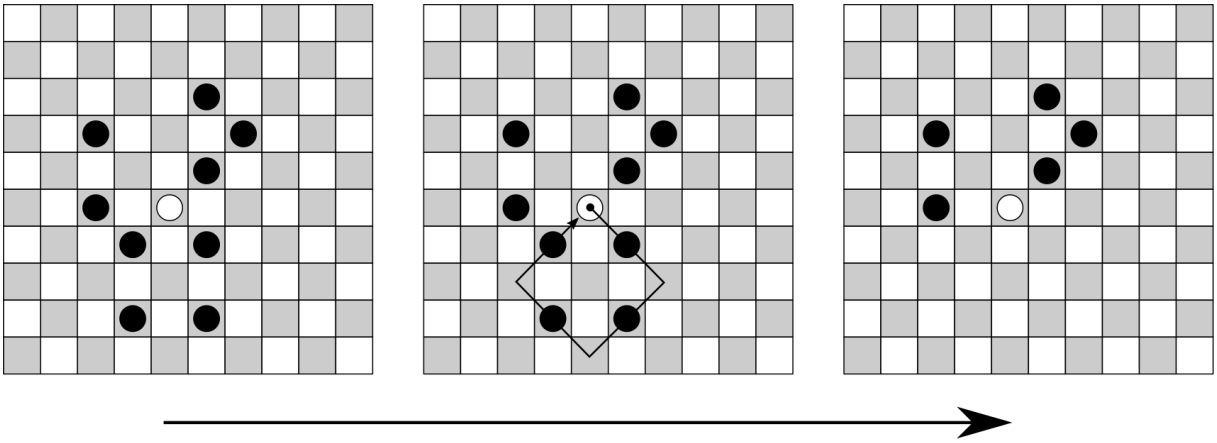
Checkers

The game of checkers is a two-player game played on a board with 10x10 squares, half of which are black and half of which are white, similar to a chessboard. Each player controls a set of pieces, and only the black squares can be occupied by the pieces. The players take turns moving their pieces, with the goal of capturing their opponent's pieces.

We are concern about capturing moves, that is a move in which a player's piece jumps over an adjacent opponent's piece and lands on the empty square immediately behind it. The captured piece is then removed from the board. It is important to note that capturing moves can only be made diagonally and cannot be made over an empty square or one of the player's own pieces. Additionally, a player must make a capturing move if one is available; if multiple capturing moves are available, the player can choose which one to make. Consecutive captures are also possible if they are made with a single piece. It is legal to make both forward and backward jumps.



Given a specific position in a game of checkers, where it is currently the turn of the player controlling the light pieces, the objective is to determine the highest number of opponent's dark pieces that can be captured in the upcoming move.

Input

The first line of input contains the integer *T*, representing the number of test cases. Each test case begins with an empty line, followed by a 10x10 board consisting of 10 lines, each containing 10 characters. The "#" and "." characters represent empty black and white squares, while "W" and "B" denote squares occupied by light and dark pieces, respectively.

Output

The expected output for each test case is a single line containing the highest number of captures possible. If no legal moves are available (such as when there are no light pieces on the board), the output should be 0.

Sample Input

```
2

.#.#.#.#.#
#.#.#.#.#.
.#.#.B.#.#
#.#.#.#.#.
.#.#.B.#.#
#.#.W.#.#.
.#.#.#.#.#
#.#.#.B.#.
.#.#.#.#.#
#.#.#.#.#.

.#.#.#.#.#
#.#.#.#.#.
.#.#.B.#.#
#.B.#.B.#.
.#.#.B.#.#
#.B.W.#.#.
.#.B.B.#.#
#.#.#.#.#.
.#.B.B.#.#
#.#.#.#.#.
```

Sample Output

```
2
4
```