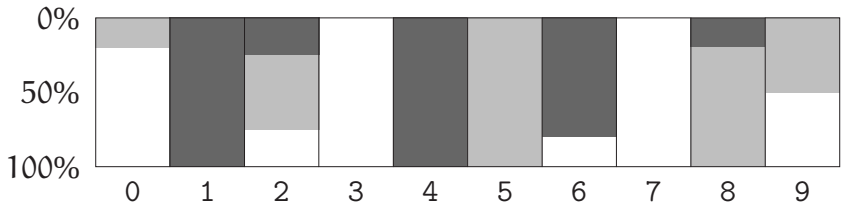
Shades

Computer Science Society
Programming Contest
Fall 2011

Just 24 hours before the contest, I was sitting on the fourth floor of the University Library trying to think of one more problem. I looked over at a long row of ten picture windows with shades pulled down various distances, and I wondered how much light they let in. Then I thought, that's too easy, let's imagine each window has two shades—one opaque and one translucent—that can be independently pulled down various distances:



Each shade can be pulled down from the top, covering some whole percentage (between 0% and 100%) of a window. An opaque shade passes no light, a translucent shade passes half of all light, and an unshaded window passes all light. Full light occurs when all shades are completely up. Starting with all shades completely up, a sequence of shade settings is given, and you will compute the percentage of full light that the ten windows collectively pass after each setting.

Input Format

Each line of input describes a shade setting, consisting of a window number $0 \leqslant w \leqslant 9$, the word `opaque` or `translucent`, and a whole percentage (between 0 and 100) of the window that the shade setting will cover.

Output Format

Starting with all shades completely up, compute and print the percentage (to the nearest whole number) of full light that the ten windows collectively pass after each setting, as shown in the sample output below.

Input Sample

```
0 translucent 20
1 opaque 100
2 opaque 25
2 translucent 75
4 opaque 100
5 translucent 100
6 opaque 80
8 opaque 20
8 translucent 100
9 translucent 50
```

Output Sample

```
99
89
87
84
74
69
61
59
55
53
```