Lineup

On June 13th team Germany has its first match in the FIFA world cup against team Australia. As the coach of team Germany, it is your duty to select the lineup for the game. Given this is your first game in the cup, naturally you want to make a good impression. Therefore you'd like to play with the strongest lineup possible.

You have already decided on the tactical formation you wish to use, so now you need to select the players who should fill each of the *11* positions in the team. Your assistant has selected the *11* strongest players from your squad, but this still leaves the question where to put which player.

Most players have a favoured position on the field where they are strongest, but some players are proficient in different positions. Your assistant has rated the playing strength of each of your *11* players in each of the *11* available positions in your formation, where a score of *100* means that this is an ideal position for the player and a score of *0* means that the player is not suitable for that position at all. Find the lineup which maximises the sum of the playing strengths of your players for the positions you assigned them. All positions must be occupied, however, do not put players in positions they are not proficient with (i.e.\ have a score of *0*).

Input

The input consists of several test cases. The first line of input contains the number C of test cases. For each case you are given *11* lines, one for each player, where the i-th line contains *11* integer numbers s_{ij} between *0* and *100*. s_{ij} describes the i-th player's strength on the j-th position. No player will be proficient in more than five different positions.

Output

For each test case output the maximum of the sum of player strengths over all possible lineups. Each test case result should go on a separate line. There will always be at least one valid lineup.

Example

Input:
```
1
100 0 0 0 0 0 0 0 0 0 0
0 80 70 70 60 0 0 0 0 0 0
0 40 90 90 40 0 0 0 0 0 0
0 40 85 85 33 0 0 0 0 0 0
0 70 60 60 85 0 0 0 0 0 0
0 0 0 0 0 95 70 60 60 0 0
0 45 0 0 0 80 90 50 70 0 0
0 0 0 0 0 40 90 90 40 70 0
0 0 0 0 0 0 50 70 85 50 0
0 0 0 0 0 0 66 60 0 80 80
0 0 0 0 0 0 50 50 0 90 88
```

Output: