You are given an unlimited number of pebbles to distribute across an $N \times N$ game board ($1 \leq N \leq 15$), where each square on the board contains an integer point value between 1 and 99, inclusive. The integers on a given board may not be unique.

A 6×6 board might look like this:

$$
\begin{array}{cccccc}
78 & 78 & 11 & \mathbf{55} & 20 & 11 \\
98 & 54 & 81 & 43 & 39 & 97 \\
12 & 15 & 79 & 99 & 58 & 10 \\
\mathbf{13} & 79 & 83 & 65 & 34 & \mathbf{17} \\
85 & 59 & 61 & 12 & 58 & 97 \\
40 & 63 & 97 & \mathbf{85} & 66 & 90 \\
\end{array}
$$

The player distributes pebbles across the board so that:

- At most one pebble resides in any given square.

- No two pebbles are placed on adjacent squares. Two squares are considered adjacent if they are horizontal, vertical, or diagonal neighbors. There is no board wrap-in the above example, 55 and 85 are not neighbors, nor are 13 and 17.

The goal is to maximize the number of points claimed by your placement of pebbles.

Input

Input to your program will be a sequence of boards, terminated by end-of-file. Each board will be separated from the next by a blank line. Each board is a sequence of N lines consisting of N integers separated from each other by whitespace.

Output

For each board, your program is to print a line containing the maximum number of points attainable. The result is to begin in the first column with no leading zeroes or trailing whitespace.

Sample Input

```
71 24 95 56 54
85 50 74 94 28
92 96 23 71 10
23 61 31 30 46
64 33 32 95 89

78 78 11 55 20 11
98 54 81 43 39 97
12 15 79 99 58 10
13 79 83 65 34 17
85 59 61 12 58 97
40 63 97 85 66 90

15 95 24 35 79 35 55 66 91 95 86 87
94 15 84 42 88 83 64 50 22 99 13 32
85 12 43 39 41 23 35 97 54 98 18 85
84 61 77 96 49 38 75 95 16 71 22 14
18 72 97 94 43 18 59 78 33 80 68 59
26 94 78 87 78 92 59 83 26 88 91 91
34 84 53 98 83 49 60 11 55 17 51 75
29 80 14 79 15 18 94 39 69 24 93 41
66 64 88 82 21 56 16 41 57 74 51 79
49 15 59 21 37 27 78 41 38 82 19 62
54 91 47 29 38 67 52 92 81 99 11 27
31 62 32 97 42 93 43 79 88 44 54 48
```

Sample Output

```
572
683
2755
```