

B、 n^3 过一万

时间限制3S，内存限制1024M，B.in/out

你有一个数组 $[1, 2, \dots, n]$ ，其中 n 是偶数。

每次你可以选择两个相邻的元素删除，如果这两个元素为 i 和 j ，那么代价为 $cost(i, j)$ 。

你可以用 $n/2$ 轮将所有元素删完，问这 $n/2$ 次删除操作代价的最大值最小是多少？

输入格式

第一行，包含一个整数 n 。

为了减小输入的大小，接下来一行包含 $n^2/4$ 个数字，表示

$cost(1, 2), cost(1, 4), \dots, cost(1, n), cost(2, 3), cost(2, 5), \dots, cost(2, n-1), \dots, cost(n-1, n)$ 。

因为如果两个元素奇偶性相同，那么显然不可能在一次里面被删除。这里将所有奇偶性不同的 (i, j) 按 i 为第一关键字 j 为第二关键字的顺序排列，并且依次输入 $cost(i, j)$ 的值。

输出格式

一行，输出将所有元素删除的代价。

样例输入1

```
10
20 21 2 11 25 3 24 18 8 6 17 7 5 22 4 23 14 15 1 19 16 12 10 13 9
```

样例输出1

```
14
```

样例输入输出2

见附加文件。

数据规模

对于20%的数据， $n \leq 10$ 。

对于接下来的数据， $n = 500, 1000, 1500, 2000, 2500, 3000, 4000, 4000$ 。

对于100%的数据，保证输入的 $cost$ 是1到 $n^2/4$ 的排列。

注意输入时间。