B
Coin Collecting

As a member of the Association of Coin Minters (ACM), you are fascinated by all kinds of coins and one of your hobbies involves collecting national currency from different countries. Your friend, also an avid coin collector, has her heart set on some of your precious coins and has proposed to play a game that will allow the winner to acquire the loser's collection (or part thereof).

She begins by preparing two envelopes, each of them enclosing two coins that come from different countries. Then she asks you to choose one of the two envelopes. You can see their contents before making your choice, and also decline the offer and take neither. This process is repeated for a total of r times. As the game progresses, you are also allowed to change your mind about your previous picks if you think you can do better. Eventually, your friend examines the envelopes in your final selection, and from among them she picks a few envelopes herself. If her selection is non-empty and includes an even number of coins from every country (possibly zero), she wins and you must hand over your entire coin collection to her, which would make years of painstaking effort go to waste and force you to start afresh. But if you win, you get to keep the coins from all the envelopes you picked.

Despite the risks involved, the prospect of enlarging your collection is so appealing that you decide to take the challenge. You'd better make sure you win as many coins as possible.

Input

The first line of each test case is the number r of rounds ($1 \leq r \leq 300$); a line with $r = 0$ indicates the end of the input. The next r lines contain four non-negative integers $0 \leq a, b, c, d < 10,000$, meaning that your friend puts coins from countries labelled a and b inside one of the envelopes, and c and d inside the other one. A blank line separates test cases.

Output

Print a line per test case containing the largest number of coins you are guaranteed.

Sample Input

```
4
0 1 0 5
5 1 0 5
1 2 0 1
1 5 2 0

6
1 4 1 4
2 4 2 4
0 3 0 3
0 4 0 4
4 3 4 3
1 3 1 3

0
```

Sample Output

```
6
8
```