

Problem 3—Domino Chains

Cersei Lannister, in order to demonstrate her power, enjoys making elaborate domino structures and knocking them down. However, she sometimes enjoys playing with her dominos to make long chains.

A domino contains two non-negative integers, one on the left and one on the right. For the purposes of this problem, a domino *cannot* be flipped around. The left integer is always on the left and the right integer is always on the right. A domino chain occurs when dominoes are laid end to end so that the right integer of a domino (other than the one on the right end of the chain) matches the left integer of the domino immediately to its right. Given a collection of dominoes, you are to determine the length of the longest chain that can be built from this collection.

INPUT SPECIFICATION. Each input case begins with the number of dominoes followed by <EOLN>. Following this are the dominoes themselves. Each domino is represented by the left integer and the right integer separated by one space. The dominoes themselves are also separated by one space. The list of dominoes is terminated by <EOLN>. “0<EOLN>” follows the last data case. The 0 is not to be processed; it merely signifies the end of input.

OUTPUT SPECIFICATION. The output cases are to be processed in the same order as the input cases. Each case will be of the form “Case c : The longest chain contains d dominoes.” where c is the case number, and d is the length of the longest chain that can be constructed from that set. Two <EOLN> characters should follow each case.

SAMPLE INPUT.

```
5<EOLN>
4 5 3 4 2 3 1 2 0 1<EOLN>
5<EOLN>
1 1 2 2 3 3 4 4 5 5<EOLN>
0<EOLN>
<EOF>
```

SAMPLE OUTPUT.

```
Case 1: The longest chain contains 5 dominoes.<EOLN>
<EOLN>
Case 2: The longest chain contains 1 dominoes.<EOLN>
<EOLN>
<EOF>
```