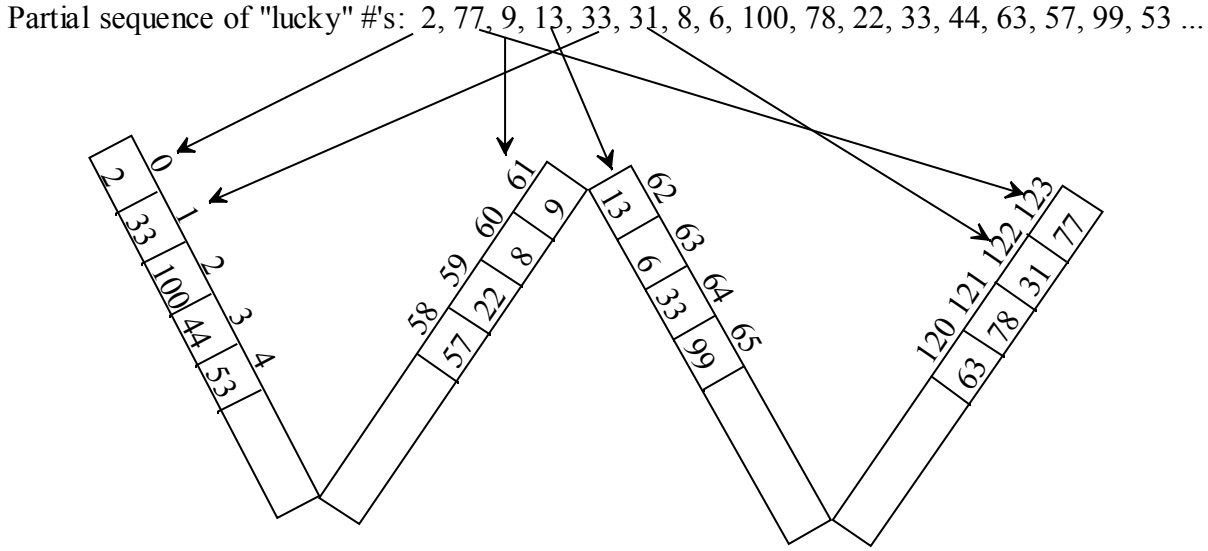
Problem 6—"W"inning Sort

Professor Plum likes to buy one ticket to a daily lottery game called Pick-4. In Pick-4 you pick four unique numbers between 1 and 1000, and the computer randomly picks four unique numbers between 1 and 1000. You win if all four of your numbers match the computers in the exact order. Needless to say, you don't win often.

Professor Plum has an idea for picking his Pick-4 numbers for every day of the month using a "W" sort ("W" for "W"inning). Before each month starts he picked his luckiest 124 unique numbers between 1 and 1000, then arranged them into a sequence from most-lucky to least-lucky. He wants you to write a program to perform a "W" sort on this sequence of numbers to determine the Pick-4 numbers for every day of the month.

A "W" sort is easily described by a diagram. The array is thought of as four "legs" of the W with each leg filled from top to bottom. The sequence of lucky numbers is scanned from most-lucky to "least-lucky" with the legs filled in the order: first leg, fourth leg, second leg and third leg before repeating.



After the "W" sort, the Pick-4 numbers for every day of the month are found by scanning the array from index 0 to index 123 with groups of four numbers being a Pick-4. In the above example the 1st of the month's Pick-4 numbers are from indices 0 to 3 (i.e., Pick-4 of 2 33 100 44), the 2nd of the month's Pick-4 numbers are from indices 4 to 7, the 3rd of the month's Pick-4 numbers are from indices 8 to 11, etc. The 31st of the month's Pick-4 numbers are from indices 120 to 123 (i.e., Pick-4 numbers of 63, 78, 31, 77).

Input Format

A single line of input contains the sequence of 124 lucky numbers ordered from most to least lucky.

Output Format

The Pick-4 numbers ordered by day of the month. Each line contains the day of the month (1-31), a colon (':') and the four Pick-4 numbers each proceeded by a space.

Input Sample (partial – full input would have 124 integers)

```
2 77 9 13 33 31 8 6 100 78 22 33 44 63 57 99 53 … <remaining omitted>
```

Output Sample (partial – full output would have 31 lines for each day of the month)

```
1: 2 9 13 77
```

```
...
16: 8 9 13 6
...
31: 63 78 31 77
```