Problem 2	**Blockbuster Connections**	11 Points

Program Name: block.cpp **Input File: block.dat**

There is a television gameshow called "Blockbusters" in which opposing contestants try to answer questions building a path across a board to win a game. See the Samples of the board in Figure 1 below. A "Family Pair" composed of two relatives must build a contiguous path from left to right before a "Solo Player" builds a contiguous path from top to bottom. The contestants build a path by correctly responding to a question whose answer begins with the letter in the selected box. Every question is open to all players (who buzz in).

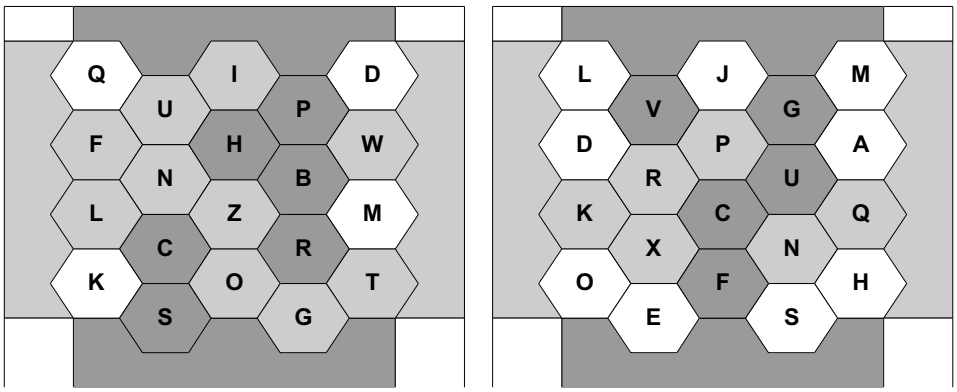

Figure 1: A Family Pair Win (Left) and a Solo Player Win (Right)

Your job is to write software that, given a board layout and a list of letters correctly answered by the contestants, determines if

1. The Solo Player Wins
2. The Family Pair Wins
3. No Winner Yet

You can see from the above examples, that the Family Pair won the game on the left with the Path LNZOGT. (In this case, other winning paths exist for the Family Pair.) The Solo Player won the game on the right with the Path GUCF. Note that not all blocks that are won need be used in the path to win. Further, the nature of the board prevents a path from top-to-bottom and a path from left-to-right from existing at the same time. Therefore, there can be only one winning path.

Your program will read in a board configuration as a single line of 20 characters (upper case alphabetic) where the order of the characters is in the order seen in Figure 2. No character is used twice and (obviously) six of the characters are unused. The board configuration component of the input line is terminated by a single colon.

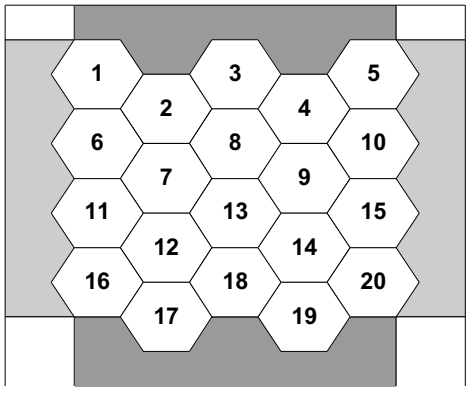

Figure 2: Location of the Input Characters on the Blockbuster Board

You program will then read a series of up to 20 characters which are the results of the contested questions to "test" the board for a winner. The Solo Player's blocks are in lower case; the Family Pair's blocks are in upper case. You are guaranteed that no character appears more than once and that all characters that do appear in the contested question results can be found on the board configuration. The end of the "test" will always end with single colon.

From Figure 1, the two input lines would be

```
QUIPDFNHBWLCZRMKSOGT:scNLFUhIWpbrGTOZ:
LVJGMDRPUAKXCNQOEFSH:vKXQfcPRNug:
```

And these two input lines would result in the output
```
"Family Pair Wins"
"Solo Player Wins"
```

But it is possible that your program will find that no one has won yet as is the case with the line
```
QUIPDFNHBWLCZRMKSOGT:scNLFUhIWpbrGT:
```

In which case your program would print the output
```
"No Winner Yet"
```

Input
Input to your program consists of a series of lines each of which contain a single test. Each input line contains 20 alphabetic characters prescribing the board configuration as described above. Column 21 contains a single colon. The line then contains between zero and 20 characters prescribing the results of the contested questions. The last column on the line contains a single colon.

Your program should continue reading test cases until it reaches the end-of-file. The file will not contain any extraneous or erroneous input (such as blank lines, embedded spaces, invalid characters).

Output
For each line of input, your program should print one of the following lines of output on a line by itself to the screen: `"Family Pair Wins", "Solo Player Wins", or "No Winner Yet"`. Your output should not print the quotations nor any other extraneous output and the output results should start in column 1.

Example: Input File
```
QUIPDFNHBWLCZRMKSOGT:scNLFUhIWpbrGTOZ:
LVJGMDRPUAKXCNQOEFSH:vKXQfcPRNug:
QUIPDFNHBWLCZRMKSOGT:scNLFUhIWpbrGT:
```
Output to screen
```
Family Pair Wins
Solo Player Wins
No Winner Yet
```