Coding exercise

As we discussed, here is a small problem that we would like you to tackle.

We do not expect it to take more than a couple of hours to work on, so we don't expect anything too elaborate. If you want to build a distributed calculator deployed on the cloud then go for it, but a small command-line program is great too. We just want to be able to gauge how you approach developing software and, should you be successful, have something to discuss when we meet.

You can choose any language you are comfortable with, although we would expect most candidates to use a modern OO or functional language. The only guideline we have is that your code

- should be easy to compile & run (please provide instructions)
- reads data from `stdin` and outputs to `stdout`
- can demonstrably produce the expected output

Building a betting host

You have been asked to write a program to calculate the dividends for a simplified form of Tote betting for Tabcorp.

Tote betting involves punters choosing the outcome of a race by placing bets into a pool of money. Punters who successfully predict the outcome of a race take a share of the pool proportional to their stake. For example, a punter who places a `$2` bet on a winning selection would receive twice the winnings of a punter who placed a `$1` stake. Tabcorp takes a commission out of the pool before it is split between winning punters.

The calculator must support three products with the following rules:

WIN

- Punters must choose the winner of a race
- Tabcorp takes a `15%` commission from the `Win` pool
- The remaining total is split, proportionally to stake, amongst punters who chose the correct winning horse.

PLACE

- Punters must choose the first, second or third place horse in a race
- Tabcorp takes a `12%` commission from the `Place` pool
- The total pool is split evenly into 3 and each of these amounts is then split, proportionally to stake, amongst the punters who chose each placed horse

EXACTA

- Punters must choose the first and second place runners in a race in the correct order
- Tabcorp takes an `18%` commission from the `Exacta` pool
- The remaining total is split, proportionally to stake, amongst punters who chose the correct first and second horse in correct order

After a race has been run, Tabcorp publishes the dividends for each product. This is the return for a `$1` stake for each paying selection in the race. All dividends are calculated to the nearest `$0.01`.

Input: list of bets

Your program should read its input from `stdin`. It will receive a list of bets placed on the current race, one per

line. The format of bets is `Bet:<product>:<selections>:<stake>`, where

- `<product>` is one of `W`, `P`, `E`
- `<selection>` is either a single runner number (e.g. `4`) for Win and Place, or two runner numbers (e.g. `4,3`) for Exacta and Quinella
- `<stake>` is an amount in whole dollars (e.g. `35`)

For example:

- `Bet:W:3:5` is a `$5` bet on horse 3 to win
- `Bet:P:2:10` is a `$10` bet on horse 2 to come first, second or third
- `Bet:E:5,7:15` is a `$15` bet on horses 5 and 7 to come first and second in that order

Input: race results

Eventually, your program will receive the race results on `stdin`. You can assume this is the last input it will receive.

The format of the results is `Result:<first>:<second>:<third>`.

For example, `Result:5:3:8` represents a race where horse 5 finished first, horse 3 finished second and horse 8 finished third.

Output: dividends

When bets have been placed and results provided, your program should generate the dividends for each product for a race. The dividends should be printed on `stdout`, in the following format: `<product>:`

`<winningSelections>:<dividend>`.

For example:

```
W:2:$2.61     # Win bet on horse 2 yields $2.61
P:2:$1.06     # Place bet on horse 2 yields $1.06
P:3:$1.27     # Place bet on horse 3 yields $1.27
P:1:$2.13     # Place bet on horse 1 yields $2.13
E:2,3:$2.43   # Exacta on horses 2,3 yields $2.43
```

Example

Here is sample data for a given race. This is what your program will receive on `stdin`:

```
Bet:W:1:3
Bet:W:2:4
Bet:W:3:5
Bet:W:4:5
Bet:W:1:16
Bet:W:2:8
Bet:W:3:22
Bet:W:4:57
Bet:W:1:42
Bet:W:2:98
Bet:W:3:63
Bet:W:4:15
Bet:P:1:31
Bet:P:2:89
Bet:P:3:28
```

```
Bet:P:4:72
Bet:P:1:40
Bet:P:2:16
Bet:P:3:82
Bet:P:4:52
Bet:P:1:18
Bet:P:2:74
Bet:P:3:39
Bet:P:4:105
Bet:E:1,2:13
Bet:E:2,3:98
Bet:E:1,3:82
Bet:E:3,2:27
Bet:E:1,2:5
Bet:E:2,3:61
Bet:E:1,3:28
Bet:E:3,2:25
Bet:E:1,2:81
Bet:E:2,3:47
Bet:E:1,3:93
Bet:E:3,2:51
Result:2:3:1
```

And this is what it should output on `stdout`:

```
Win:2:$2.61
Place:2:$1.06
Place:3:$1.27
Place:1:$2.13
Exacta:2,3:$2.43
```