Problem B: Overflowing Bookshelf

Source file: `bookshelf.{c, cpp, java}`

Input file: `bookshelf.in`

Agnes C. Mulligan is a fanatical bibliophile – she is constantly buying new books, and trying to find space for those books. In particular, she has a shelf for her "to be read" books, where she puts her newest books. When she decides to read one of these books, she removes it from the shelf, making space for more books. Sometimes, however, she buys a new book and puts it on the shelf, but because of limited space, this pushes one or more books off the shelf at the other end. She always adds books on the left side of the shelf, making books fall off the right side. Of course, she can remove a book from any location on the shelf when she wants to read one.

Your task will be to write a simulator that will keep track of books added and removed from a shelf. At the end of the simulation, display the books remaining on the shelf, in order from left to right. Books in each simulation will be identified by a unique, positive integer, $0 < I \leq 100$. There are three types of events in the simulation:

- `Add`: A new book is pushed on the left end of the shelf, pushing other books to the right as needed. No book moves to the right unless it is pushed by an adjacent (touching) book on its left. Any book that is not *entirely* on the shelf falls off the right edge. No single book will ever be wider than the given shelf. No book that is currently on the shelf will be added again.
- `Remove`: If the book is on the shelf, then the book is removed from the shelf, leaving a hole. If the book isn't on the shelf, the operation is ignored.
- `End`: End the simulation for this case and print the contents of the shelf.

Input: The input file will contain data for one or more simulations. The end of the input is signalled by a line containing -1. Each simulation will begin with the integer width of the shelf, s, $5 \leq s \leq 100$, followed by a series of `add` and `remove` events. An `add` event is a single line beginning with an upper case 'A' followed by the book ID, followed by the integer width of the book, w, $0 < w \leq s$. A `remove` event is a single line beginning with an upper case 'R' followed by the the book ID. Finally, the `end` event is a line containing only a single upper case 'E'. Each number in an event is preceded by a single blank.

Output: For each simulation case, print a single line containing a label (as shown in the output sample), followed by the list of IDs of books remaining on the shelf, in order from left to right.

Example input:	Example output:
10 R 3 A 6 5 A 42 3 A 3 5 A 16 2 A 15 1 R 16 E 7 A 49 6 A 48 2 R 48 E 5 A 1 1 A 2 1 A 3 1 R 2 A 4 1 A 5 1 R 5 R 4 A 6 1 A 7 4 E -1	PROBLEM 1: 15 3 PROBLEM 2: PROBLEM 3: 7 6

Last modified on October 30, 2005 at 12:10 PM.