```
 1 /**
 2  * Controller interface.
 3  *
 4  * @author Bruce W. Weide
 5  *
 6  * @mathmodel <pre>
 7  * type NNCalcController is modeled by
 8  *    (model: NNCalcModel,
 9  *     view: NNCalcView)
10  * </pre>
11  * @initially <pre>
12  * (NNCalcModel model, NNCalcView view):
13  *    ensures
14  *       this.model = model  and
15  *       this.view = view
16  * </pre>
17  */
18 public interface NNCalcController {
19
20     /**
21      * Processes event to clear bottom operand.
22      *
23      * @updates this.model.bottom, this.view
24      * @ensures <pre>
25      * this.model.bottom = 0  and
26      * [this.view has been updated to match this.model]
27      * </pre>
28      */
29     void processClearEvent();
30
31     /**
32      * Processes event to swap operands.
33      *
34      * @updates this.model, this.view
35      * @ensures <pre>
36      * this.model.top = #this.model.bottom  and
37      * this.model.bottom = #this.model.top  and
38      * [this.view has been updated to match this.model]
39      * </pre>
40      */
41     void processSwapEvent();
42
43     /**
44      * Processes event to enter bottom operand to top.
45      *
46      * @updates this.model.top, this.view
47      * @ensures <pre>
48      * this.model.top = this.model.bottom  and
49      * [this.view has been updated to match this.model]
50      * </pre>
51      */
52     void processEnterEvent();
53
54     /**
55      * Processes event to do an add operation.
56      *
57      * @updates this.model, this.view
```

```
 58        * @ensures <pre>
 59        * this.model.top = 0  and
 60        * this.model.bottom = #this.model.top + #this.model.bottom  and
 61        * [this.view has been updated to match this.model]
 62        * </pre>
 63        */
 64       void processAddEvent();
 65
 66       /**
 67        * Processes event to do a subtract operation.
 68        *
 69        * @updates this.model, this.view
 70        * @requires this.model.bottom <= this.model.top
 71        * @ensures <pre>
 72        * this.model.top = 0  and
 73        * this.model.bottom = #this.model.top - #this.model.bottom  and
 74        * [this.view has been updated to match this.model]
 75        * </pre>
 76        */
 77       void processSubtractEvent();
 78
 79       /**
 80        * Processes event to do a multiply operation.
 81        *
 82        * @updates this.model, this.view
 83        * @ensures <pre>
 84        * this.mode.top = 0  and
 85        * this.model.bottom = #this.model.top * #this.model.bottom  and
 86        * [this.view has been updated to match this.model]
 87        * </pre>
 88        */
 89       void processMultiplyEvent();
 90
 91       /**
 92        * Processes event to do a divide operation.
 93        *
 94        * @updates this.model, this.view
 95        * @requires this.model.bottom > 0
 96        * @ensures <pre>
 97        * #this.model.top =
 98        *    this.model.bottom * #this.model.bottom + this.model.top  and
 99        * 0 <= this.model.top < #this.model.bottom  and
100        * [this.view has been updated to match this.model]
101        * </pre>
102        */
103       void processDivideEvent();
104
105       /**
106        * Processes event to do a power operation.
107        *
108        * @updates this.model, this.view
109        * @requires this.model.bottom <= INT_LIMIT
110        * @ensures <pre>
111        * this.model.top = 0  and
112        * this.model.bottom = #this.model.top ^ (#this.model.bottom)  and
113        * [this.view has been updated to match this.model]
114        * </pre>
```

```
115      */
116     void processPowerEvent();
117
118     /**
119      * Processes event to do a root operation.
120      *
121      * @updates this.model, this.view
122      * @requires 2 <= this.model.bottom <= INT_LIMIT
123      * @ensures <pre>
124      * this.model.top = 0  and
125      * this.model.bottom =
126      *   [the floor of the #this.model.bottom root of #this.model.top]  and
127      * [this.view has been updated to match this.model]
128      * </pre>
129      */
130     void processRootEvent();
131
132     /**
133      * Processes event to add a new (low-order) digit to the bottom operand.
134      *
135      * @param digit
136      *            the low-order digit to be added
137      *
138      * @updates this.model.bottom, this.view
139      * @requires 0 <= digit < 10
140      * @ensures <pre>
141      * this.model.bottom = #this.model.bottom * 10 + digit  and
142      * [this.view has been updated to match this.model]
143      * </pre>
144      */
145     void processAddNewDigitEvent(int digit);
146
147 }
148
```