```
1 import components.naturalnumber.NaturalNumber;
2 import components.naturalnumber.NaturalNumberSecondary;
3
4 /**
5 * {@code NaturalNumber} represented as a {@code String} with
  implementations of
6 * primary methods.
7 *
8 * @convention 
9 * [all characters of $this.rep are '0' through '9'] and
10 * [$this.rep does not start with '0']
11 * 
12 * @correspondence 
13 * this = [if $this.rep = "" then 0
14 * else the decimal number whose ordinary depiction is
  $this.rep]
15 * 
16 *
17 * @author Jonny Pater and Alex Honigford
18 *
19 */
20 public class NaturalNumber3 extends NaturalNumberSecondary {
21
22
     /*
23
     * Private members
24
     */
25
26
      /**
      * Representation of {@code this}.
27
28
29
      private String rep;
30
31
32
       * Creator of initial representation.
33
      private void createNewRep() {
34
35
          this rep = "";
36
      }
37
38
      /*
39
      * Constructors
40
     */
```

```
41
42
      /**
43
       * No-argument constructor.
44
45
      public NaturalNumber3() {
46
47
           this.createNewRep();
48
49
      }
50
51
      /**
       * Constructor from {@code int}.
52
53
54
       * @param i
55
                     {@code int} to initialize from
       *
56
       */
57
      public NaturalNumber3(int i) {
           assert i >= 0 : "Violation of: i >= 0";
58
59
           this.createNewRep();
           if (i > 0) {
60
61
               this.rep = this.rep + i;
62
           }
      }
63
64
65
66
       * Constructor from {@code String}.
67
68
       * @param s
69
                     {@code String} to initialize from
70
       */
      public NaturalNumber3(String s) {
71
72
           assert s != null : "Violation of: s is not null";
          assert s.matches("0|[1-9]\\d*") : ""
73
74
                   + "Violation of: there exists n: NATURAL (s =
  TO_STRING(n))";
           this.createNewRep();
75
76
           if (s.charAt(0) != '0') {
77
               this.rep = s;
78
           }
79
80
      }
81
82
       * Constructor from {@code NaturalNumber}.
83
```

```
84
85
        * @param n
                      {@code NaturalNumber} to initialize from
86
87
        */
88
       public NaturalNumber3(NaturalNumber n) {
            assert n != null : "Violation of: n is not null";
89
            this.createNewRep();
90
            String nn = n.toString();
91
            if (nn.charAt(0) != '0') {
92
93
                this.rep = nn;
94
           }
95
96
       }
97
98
       /*
99
        * Standard methods
100
        */
101
102
       @Override
103
       public final NaturalNumber newInstance() {
104
           try {
105
                return this.getClass().getConstructor().newInstance();
            } catch (ReflectiveOperationException e) {
106
107
                throw new AssertionError(
                        "Cannot construct object of type " +
108
   this.getClass());
109
            }
       }
110
111
112
       @Override
113
       public final void clear() {
           this.createNewRep();
114
115
116
117
       @Override
118
       public final void transferFrom(NaturalNumber source) {
           assert source != null : "Violation of: source is not
119
   null":
120
           assert source != this : "Violation of: source is not
   this":
121
           assert source instanceof NaturalNumber3 : ""
                    + "Violation of: source is of dynamic type
122
   NaturalNumberExample";
```

```
Tuesday, September 12, 2023, 10:28 PM
NaturalNumber3.java
123
            /*
124
             * This cast cannot fail since the assert above would have
   stopped
125
             * execution in that case.
126
             */
127
            NaturalNumber3 localSource = (NaturalNumber3) source;
            this.rep = localSource.rep;
128
129
            localSource.createNewRep();
        }
130
131
132
       /*
         * Kernel methods
133
134
        */
135
136
       @Override
137
        public final void multiplyBy10(int k) {
            assert 0 <= k : "Violation of: 0 <= k";</pre>
138
            assert k < RADIX : "Violation of: k < 10";</pre>
139
            if (!this.rep.equals("") || k > 0) {
140
141
                this.rep = this.rep + k;
            }
142
143
144
        }
145
146
       @Override
        public final int divideBy10() {
147
148
            int rem = 0;
            if (this.rep.length() > 0) {
149
                rem = Character
150
151
                         .getNumericValue(this.rep.charAt(this.rep.leng)
   th() - 1));
152
                this.rep = this.rep.substring(0, this.rep.length() -
   1);
153
154
            return rem;
        }
155
156
157
       @Override
158
        public final boolean isZero() {
159
            return this rep equals("");
160
        }
161
162 }
```

NaturalNumber3.java

Tuesday, September 12, 2023, 10:28 PM

163