```cpp
1
2  // Problem Set 2, 2022
3
4  #include <iostream>
5  #include <stdexcept>
6  #include<array>
7  using namespace std;
8
9  #define P1
10 #define P2
11 #define P3
12
13 #ifdef P1
14
15 #include "IntVector.h"
16
17 void runP1()
18 {
19     int lArray[] = { 34, 65, 890, 86, 16, 218, 20, 49, 2, 29 };
20     size_t lArrayLength = sizeof(lArray) / sizeof(int);
21
22     IntVector lVector( lArray, lArrayLength );
23
24     cout << "Test range check:" << endl;
25
26     try
27     {
28         int lValue = lVector[lArrayLength];
29
30         cerr << "Error, you should not see " << lValue << " here!" << endl;
31     }
32     catch (out_of_range e)
33     {
34         cerr << "Properly caught error: " << e.what() << endl;
35     }
36     catch (...)
37     {
38         cerr << "This message must not be printed!" << endl;
39     }
40
41     cout << "Test swap:" << endl;
42
43     try
44     {
45         cout << "lVector[3] = " << lVector[3] << endl;
46         cout << "lVector[6] = " << lVector[6] << endl;
47
48         lVector.swap( 3, 6 );
49
```

```cpp
50              cout << "lVector.get( 3 ) = " << lVector.get( 3 ) << endl;
51              cout << "lVector.get( 6 ) = " << lVector.get( 6 ) << endl;
52
53              lVector.swap( 5, 20 );
54
55              cerr << "Error, you should not see this message!" << endl;
56          }
57      catch (out_of_range e)
58      {
59          cerr << "Properly caught error: " << e.what() << endl;
60      }
61      catch (...)
62      {
63          cerr << "Error, this message must not be printed!" << endl; //this is ⮐
                printing
64      }
65  }
66
67  #endif
68
69  #ifdef P2
70
71  #include "SortableIntVector.h"
72
73  void runP2()
74  {
75      int lArray[] = { 34, 65, 890, 86, 16, 218, 20, 49, 2, 29 };
76      size_t lArrayLength = sizeof(lArray) / sizeof(int);
77
78      SortableIntVector lVector(lArray, lArrayLength);
79
80      cout << "Bubble Sort:" << endl;
81
82      cout << "Before sorting:" << endl;
83
84      for (size_t i = 0; i < lVector.size(); i++)
85      {
86          cout << lVector[i] << ' ';
87      }
88
89      cout << endl;
90
91
92      lVector.sort([](int aLeft, int aRight) { return aLeft >= aRight; });
93
94      cout << "After sorting:" << endl;
95
96      for ( size_t i = 0; i < lVector.size(); i++ )
97          {
```

```cpp
 98             cout << lVector[i] << ' ';
 99         }
100
101         cout << endl;
102 }
103
104 #endif
105
106 #ifdef P3
107
108 #include "ShakerSortableIntVector.h"
109
110 void runP3()
111 {
112     int lArray[] = { 34, 65, 890, 86, 16, 218, 20, 49, 2, 29 };
113     size_t lArrayLength = sizeof(lArray) / sizeof(int);
114
115     ShakerSortableIntVector lVector( lArray, lArrayLength );
116
117     cout << "Cocktail Shaker Sort:" << endl;
118
119     cout << "Before sorting:" << endl;
120
121     for ( size_t i = 0; i < lVector.size(); i++ )
122     {
123         cout << lVector[i] << ' ';
124     }
125
126     cout << endl;
127
128     // sort in decreasing order
129     lVector.sort();
130     cout << "After sorting:" << endl;
131
132     for ( size_t i = 0; i < lVector.size(); i++ )
133     {
134         cout << lVector[i] << ' ';
135     }
136
137     cout << endl;
138 }
139
140 #endif
141
142 int main()
143 {
144 #ifdef P1
145
146     runP1();
```

```cpp
147
148  #endif
149
150  #ifdef P2
151
152      runP2();
153
154  #endif
155
156  #ifdef P3
157
158      runP3();
159
160  #endif
161
162      return 0;
163  }
164
```