**Source File:**   ~/2336/49/lab49.cpp
**Input:**         under control of `main` function
**Output:**        under control of `main` function
**Value:**         2

Write a function template to implement an efficient sort using a priority queue. You should begin by inserting the elements of the vector to be sorted into a priority queue. After all of the elements have been inserted into the priority queue, remove the elements from the priority queue and place them back into the vector. A sample `main` function for testing your implementation is shown in Figure 1 and a sample execution sequence is shown in Figure 2. You will need to add a target of `lab49main` to the definition of `targets1srcfile` in your `Makefile`.

```cpp
1   #include <cstdlib>
2   #include <iostream>
3   #include <vector>
4   #include <algorithm>
5   #include <chrono>
6   #include <random>
7
8   using namespace std;
9
10  template <typename T>
11  void pqSort(vector<T>& v);
12
13  #include <lab49.cpp>
14
15  const int N = 100000;
16  enum TEST_TYPE {RANDOM, ASCENDING, DESCENDING};
17
18  int main()
19  {
20    vector<int> v, w, x;
21    default_random_engine ran;
22    uniform_int_distribution<> dis; // [0..2^31 - 1]
23    TEST_TYPE testType;
24    int i;
25
26    for (testType = RANDOM;
27         testType <= DESCENDING;
28         testType = static_cast<TEST_TYPE>(testType + 1))
29    {
30      if (!v.empty())
31        v.clear();
32      switch (testType)
33      {
34        case RANDOM:
35          for (i = 0; i < N; ++i)
36            v.push_back(dis(ran));
37          cout << "Random Data:" << endl;
38          break;
```

**Figure 1.** `/usr/local/2336/src/lab49main.C` (Part 1 of 2)

```
39          case ASCENDING:
40            for (i = 0; i < N; ++i)
41              v.push_back(i);
42            cout << "Ascending Data:" << endl;
43            break;
44          case DESCENDING:
45            for (i = 0; i < N; ++i)
46              v.push_back(N - i);
47            cout << "Descending Data:" << endl;
48            break;
49        }
50        x = v;
51        sort(x.begin(), x.end());
52
53        w = v;
54        auto start = chrono::system_clock::now();
55        pqSort(w);
56        auto stop = chrono::system_clock::now();
57        cout << "PQ Sort: "
58             << chrono::duration_cast<chrono::milliseconds>(stop-start).count()
59             << "ms" << endl;
60
61        if (x != w)
62          cout << "Sort didn't work correctly" << endl;
63        cout << endl;
64      }
65    return EXIT_SUCCESS;
66  }
```

**Figure 1.** /usr/local/2336/src/lab49main.C (Part 2 of 2)

```
 1   newuser@csunix ~> cd 2336
 2   newuser@csunix ~/2336> ./getlab.ksh 49
 3     * Checking to see if a folder exists for Lab 49. . .No
 4     * Creating a folder for Lab 49
 5     * Checking to see if Lab 49 has sample input and output files. . .No
 6     * Checking to see if /usr/local/2336/src/lab49main.C exists. . .Yes
 7     * Copying file /usr/local/2336/src/lab49main.C to folder ./49
 8     * Checking to see if /usr/local/2336/include/lab49.h exists. . .No
 9     * Copying file /usr/local/2336/src/Makefile to folder ./49
10     * Adding a target of lab49main to targets1srcfile
11     * Touching file ./49/lab49.cpp
12     * Edit file ./49/lab49.cpp in Notepad++
13   newuser@csunix ~/2336> cd 49
14   newuser@csunix ~/2336/49> ls
15   Makefile  lab49.cpp  lab49main.C
16   newuser@csnew ~/2336/49> make lab49main
17   g++ -g -Wall -std=c++11 -c lab49main.C -I/usr/local/2336/include -I.
18   g++ -o lab49main lab49main.o -L/usr/local/2336/lib -lm -lbits
19   newuser@csunix ~/2336/49> ./lab49main
20   Random Data:
21   PQ Sort: 119ms
22
23   Ascending Data:
24   PQ Sort: 149ms
25
26   Descending Data:
27   PQ Sort: 100ms
28
29   newuser@csunix ~/2336/49>
```

**Figure 2.** Commands to Compile, Link, & Run Lab 49