

[◀ Return to "C++" in the classroom](#)

Process Monitor

REVIEW

CODE REVIEW 16

HISTORY

▶ src/linux_parser.cpp 9

▶ src/system.cpp 2

▶ src/process.cpp 2

▶ src/processor.cpp 1

▼ src/format.cpp 1

```
1 #include <string>
2
3 #include "format.h"
4
5 /*
6  * Pad string with leading character if string length < size_type
7  */
8 template<typename T>
9 void Format::pad(std::basic_string<T>& s, typename std::basic_string<T>::size_
10     if (n > s.length())
11         s.insert(s.begin(), n-s.length(), c);
12 }
13
14 /*
15  * Convert seconds into HH:MM:SS format
16  */
```

```

17 std::string Format::ElapsedTime(long elapsed_seconds) {
18     int hrs, mins, secs;
19     int wished_str_length = 2;
20     char pad_with = '0';
21     std::string shrs, smins, ssecs;
22     hrs = elapsed_seconds / SECONDS_IN_HOUR;
23     //result of modulo is in seconds
24     mins = (elapsed_seconds % SECONDS_IN_HOUR) / SECONDS_IN_MINUTE;
25     secs = elapsed_seconds - hrs*SECONDS_IN_HOUR - mins*SECONDS_IN_MINUTE;
26     //convert to string and pad with leading zero if length<2
27     shrs = std::to_string(hrs);
28     smins = std::to_string(mins);
29     ssecs = std::to_string(secs);
30     pad(shrs, wished_str_length, pad_with);
31     pad(smins, wished_str_length, pad_with);
32     pad(ssecs, wished_str_length, pad_with);
33     return shrs + ":" + smins + ":" + ssecs;
34 }

```



SUGGESTION

It was so lovely to see the implementation.

By the way I am also giving you an alternative implementation you might like:

```

std::string Format::Pad(string s, char c) {
    s.insert(s.begin(), 2 - s.size(), c);
    return s;
}

string Format::Time(long int time) {
    int hours = time / (60 * 60);
    int minutes = (time / 60) % 60;
    long seconds = time % 60;
    return std::string(Format::Pad(to_string(hours), '0') + ":" +
        Format::Pad(to_string(minutes), '0') + ":" +
        Format::Pad(to_string(seconds), '0'));
}

```

35

► include/ncurses_display.h 1

► src/ncurses_display.cpp

► src/main.cpp

► include/system.h

► include/processor.h

- ▶ include/process.h
- ▶ include/linux_parser.h
- ▶ include/format.h
- ▶ README.md
- ▶ Makefile
- ▶ CMakeLists.txt

RETURN TO PATH
