

[Return to "C++" in the classroom](#)[DISCUSS ON STUDENT HUB](#)

Process Monitor

REVIEW

CODE REVIEW 24

HISTORY

▸ src/linux_parser.cpp 14

▸ src/system.cpp 2

▸ src/processor.cpp 2

▸ src/process.cpp 2

▸ src/format.cpp 1

▸ Makefile 1

▸ include/ncurses_display.h 1

▼ include/linux_parser.h 1

```
1 #ifndef SYSTEM_PARSER_H
2 #define SYSTEM_PARSER_H
3
4 #include <fstream>
5 #include <regex>
6 #include <string>
7
8 namespace LinuxParser {
9 // Paths
10 const std::string kProcDirectory{"/proc/"};
11 const std::string kCmdlineFilename{"/cmdline"};
12 const std::string kCpuinfoFilename{"/cpuinfo"};
13 const std::string kStatusFilename{"/status"};
14 const std::string kStatFilename{"/stat"};
15 const std::string kUptimeFilename{"/uptime"};
16 const std::string kMeminfoFilename{"/meminfo"};
17 const std::string kVersionFilename{"/version"};
18 const std::string kOSPath{"/etc/os-release"};
19 const std::string kPasswordPath{"/etc/passwd"};
```

SUGGESTION

You should define the hardcoded value here and then use the same when you will be finding the values related to every entity as follows:

```
const string filterProcesses("processes");
const string filterRunningProcesses("procs_running");
const string filterMemTotalString("MemTotal:");
const string filterMemFreeString("MemFree:");
const string filterCpu("cpu");
const string filterUID("Uid:");
const string filterProcMem("VmData:"); // The string can be VmSize As well
```

This will help you avoid errors due to spelling mistakes which can happen in a bigger project like this one! I have also made sure that the string is of type `const` so that nobody changes its value in other file and so it is always recommended to use `const` whenever it is possible

```
20
21 // System
22 float MemoryUtilization();
23 long UpTime();
24 std::vector<int> Pids();
25 int TotalProcesses();
26 int RunningProcesses();
27 std::string OperatingSystem();
28 std::string Kernel();
```

```

29
30 // CPU
31 enum CPUStates {
32     kUser_ = 0,
33     kNice_,
34     kSystem_,
35     kIdle_,
36     kIOWait_,
37     kIRQ_,
38     kSoftIRQ_,
39     kSteal_,
40     kGuest_,
41     kGuestNice_
42 };
43 std::vector<std::string> CpuUtilization();
44 long Jiffies();
45 long ActiveJiffies();
46 long ActiveJiffies(int pid);
47 long IdleJiffies();
48
49 // Processes
50 std::string Command(int pid);
51 std::string Ram(int pid);
52 std::string Uid(int pid);
53 std::string User(int pid);
54 long int UpTime(int pid);
55 }; // namespace LinuxParser
56
57 #endif

```

▸ **src/ncurses_display.cpp**

▸ **src/main.cpp**

▸ **README.md**

▸ **include/system.h**

▸ **include/processor.h**

▸ **include/process.h**

▸ **include/format.h**

▸ **CMakeLists.txt**

Learn the [best practices for revising and resubmitting your project](#).

RETURN TO PATH