

May 24, 20 1:13

DebugPrinter.cpp

Page 1/1

```

1  //
2  // Created by rozanecm on 5/12/20.
3  //
4
5  #include <iostream>
6  #include "DebugPrinter.h"
7  #include "../signals/SignalHandler.h"
8
9  #define DEBUG_FILE_PATH "../log_info.log"
10
11 DebugPrinter::DebugPrinter(bool should_i_print) : should_i_print(
12     should_i_print) {
13     canal_debug.abrir();
14     debug_file.open(DEBUG_FILE_PATH, std::ios::out | std::ios::trunc);
15     SignalHandler::getInstance()->registrarHandler(SIGINT, &sigint_handler);
16 }
17
18 DebugPrinter::~DebugPrinter() {
19     canal_debug.cerrar();
20     canal_debug.eliminar();
21     debug_file.close();
22     SignalHandler::destruir();
23 }
24
25 void DebugPrinter::atender_debug_msgs() {
26     if (should_i_print) {
27         ssize_t bytesLeidos = 1;
28
29         bool seguir_escuchando = true;
30         while (sigint_handler.getGracefulQuit() == 0 and seguir_escuchando) {
31             bytesLeidos = canal_debug.leer(static_cast<void *>(buffer),
32                                           FIFO_DEBUG_BUFFSIZE);
33
34             if (bytesLeidos > 0) {
35                 std::string mensaje = buffer;
36                 mensaje.resize(bytesLeidos);
37                 this->print(mensaje);
38             } else {
39                 seguir_escuchando = false;
40             }
41         }
42     }
43 }
44
45 void DebugPrinter::ejercer_tarea() {
46     atender_debug_msgs();
47 }
48
49 void DebugPrinter::print(const std::string &msg) {
50     debug_file << msg << std::flush;
51 }

```