

SWINBURNE UNIVERSITY OF TECHNOLOGY

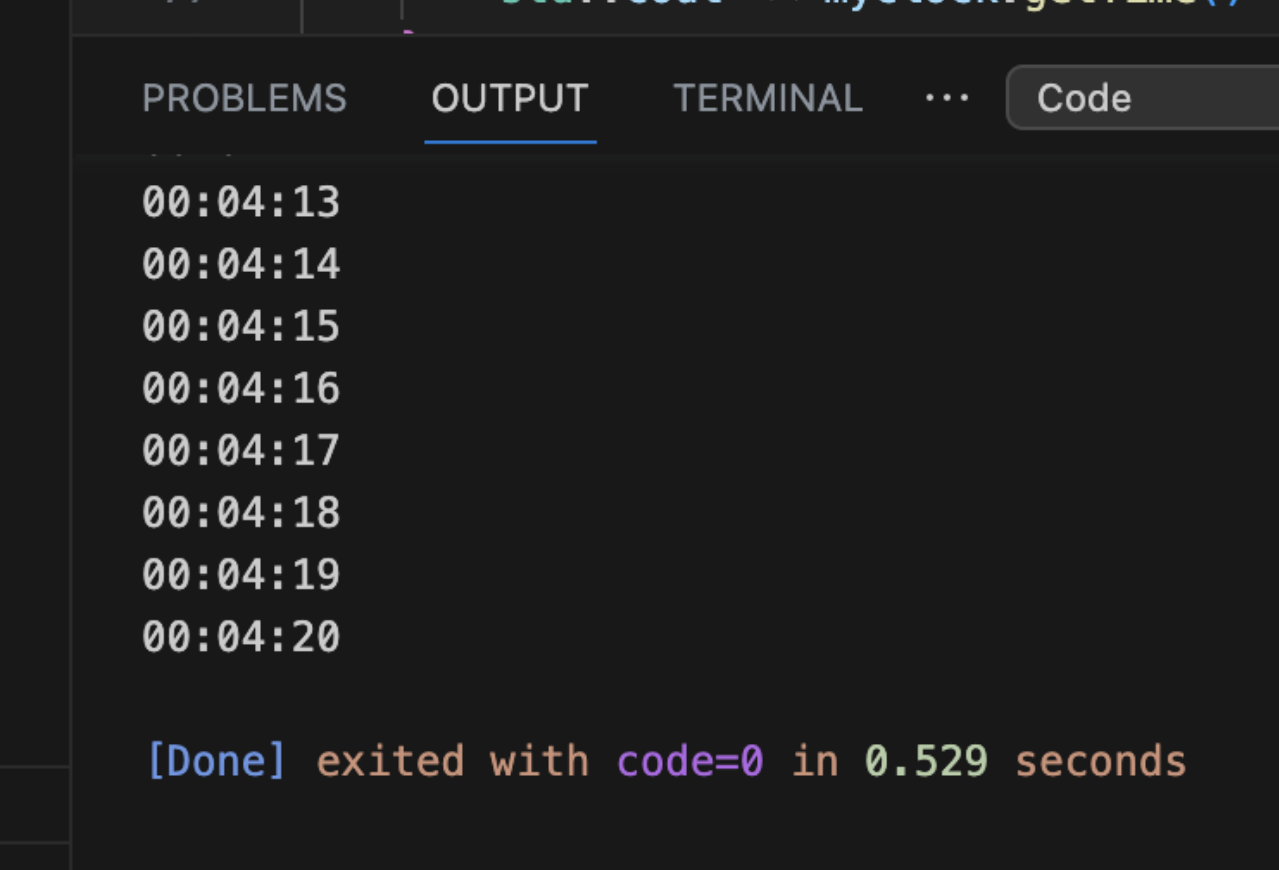
COS20007 OBJECT ORIENTED PROGRAMMING

Clock in Another Language

PDF generated at 15:15 on Tuesday 16th May, 2023

```
1  #include <iostream>
2  #include <iomanip>
3  #include <sstream>
4  #include <string>
5
6  // This class provides methods to increment, reset, and retrieve
7  class Counter {
8  private:
9      int count;
10
11 public:
12     Counter() : count(0) {}
13
14     void increment() {
15         count++;
16     }
17
18     void reset() {
19         count = 0;
20     }
21
22     int getCount() const {
23         return count;
24     }
25 };
26
27 // Keep track of time
28 class Clock {
29 private:
30     Counter seconds;
31     Counter minutes;
32     Counter hours;
33
34 public:
35     Clock() {}
36
37     void tick() {
38         seconds.increment();
39         if (seconds.getCount() > 59) {
40             minutes.increment();
41             seconds.reset();
42         }
43         if (minutes.getCount() > 59) {
44             hours.increment();
45             minutes.reset();
46         }
47         if (hours.getCount() > 23) {
48             hours.reset();
49             minutes.reset();
50             seconds.reset();
51         }
52     }
53 }
```

```
54     void resetClock() {
55         hours.reset();
56         minutes.reset();
57         seconds.reset();
58     }
59
60     //To build the formatted string representation of the time
61     //Setfill manipulator ensures that leading zeros are added
62     std::string getTime() const {
63         std::ostringstream oss;
64         oss << std::setfill('0') << std::setw(2) << hours.getCount() << ":"
65             << std::setfill('0') << std::setw(2) << minutes.getCount() << ":"
66             << std::setfill('0') << std::setw(2) << seconds.getCount();
67         return oss.str();
68     }
69 };
70
71 int main() {
72     Clock myClock;
73     myClock.resetClock();
74
75     for (int i = 0; i < 260; i++) {
76         myClock.tick();
77         std::cout << myClock.getTime() << std::endl;
78     }
79
80     return 0;
81 }
```



The screenshot shows the 'OUTPUT' tab of an IDE. The output consists of a list of timestamps from 00:04:13 to 00:04:20, followed by a completion message: [Done] exited with code=0 in 0.529 seconds. The IDE interface includes tabs for PROBLEMS, OUTPUT, and TERMINAL, with a 'Code' button on the right.

```
00:04:13
00:04:14
00:04:15
00:04:16
00:04:17
00:04:18
00:04:19
00:04:20

[Done] exited with code=0 in 0.529 seconds
```