Swinburne University of Technology

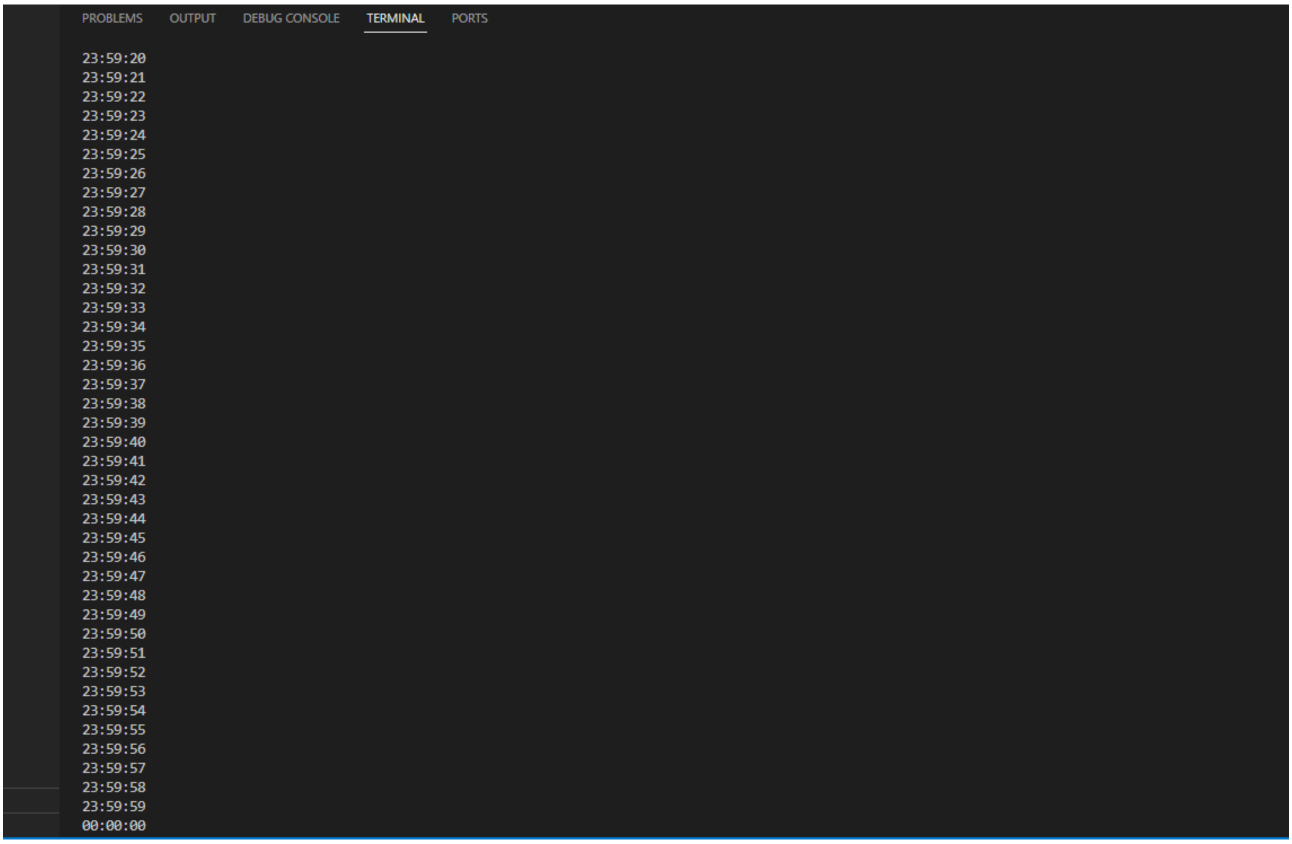
COS20007 Object Oriented Programming

Clock in Another Language

PDF generated at 17:31 on Thursday 21st September, 2023

Produced by OOP Submission Formatter

```
Counter.py
class Counter:
    def __init__(self, name):
        self._name = name
        self._count = 0
        # initialise counter with a name

    @property
    def count(self):
        return self._count

    def increment(self):
        self._count += 1
        # count value increments by 1

    def reset(self):
        self._count = 0
        # count value reset to 0



clock.py

from counter import Counter

class Clock:
    def __init__(self):
        self._hours = Counter("hours")
        self._minutes = Counter("minutes")
        self._seconds = Counter("seconds")

    def tick(self):
        self._seconds.increment()
        # increment of seconds

        if self._seconds.count == 60:
            self._seconds.reset()
            self._minutes.increment()
            # as seconds reach 60, minutes raise by 1

            if self._minutes.count == 60:
                self._minutes.reset()
                self._hours.increment()
                # as minutes reach 60, hours raise by 1

                if self._hours.count == 24:
                    self._hours.reset()
                    # clock resets once hours reach 24

    def read_time(self):
        # format time string
        display_time = "{:02d}:{:02d}:{:02d}".format(
            self._hours.count, self._minutes.count, self._seconds.count
        )
        return display_time

    def reset(self):
        self._seconds.reset()
        self._minutes.reset()
        self._hours.reset()



main.py


from clock import Clock

def main():
    clock = Clock()

    print("Digital Clock")

    # 86400 seconds = 24 hours
    for _ in range(86400):
        clock.tick()
        print(clock.read_time())

    clock.reset()


if __name__ == "__main__":
    main()
    # entry point of the program

```



```
23:59:20
23:59:21
23:59:22
23:59:23
23:59:24
23:59:25
23:59:26
23:59:27
23:59:28
23:59:29
23:59:30
23:59:31
23:59:32
23:59:33
23:59:34
23:59:35
23:59:36
23:59:37
23:59:38
23:59:39
23:59:40
23:59:41
23:59:42
23:59:43
23:59:44
23:59:45
23:59:46
23:59:47
23:59:48
23:59:49
23:59:50
23:59:51
23:59:52
23:59:53
23:59:54
23:59:55
23:59:56
23:59:57
23:59:58
23:59:59
00:00:00
```