

## Lab 1.1 - myCat

Advanced Programming

---

### Requirements:

- Rewrite the cat.c program by using read, write, open and close instead of their library equivalents.
  - Generate a report with performance metrics between the existing cat.c and your mycat.c. The format of the report is free, you can add charts or whatever helps to understand the speed of both approaches.
  - Update the README.md file with instructions about how to compile and execute your program.
- 

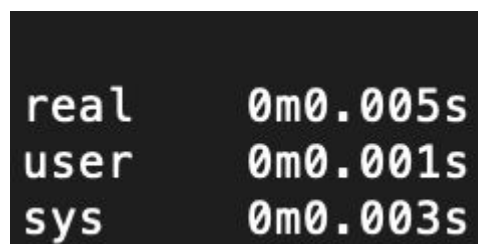
This time, we need to develop our cat function to read and write a file, also measure the performance against the already existing cat.c file

<https://github.com/CodersSquad/ap-labs/blob/master/labs/lab1.1/cat2.c>

As the requirements says, we had to rewrite the program using basic functions as read, write, open and close.


### Time comparison:

I use the `time cat file.c` command to get the following results



```
real    0m0.005s
user    0m0.001s
sys     0m0.003s
```

Img 1. Original time mycat.c file



```
real    0m0.004s
user    0m0.001s
sys     0m0.002s
```

Img 2. Original time cat2.c file

---

<b>Real</b>	The actual time spent in running the process from start to finish.
<b>User</b>	The cumulative time spent by all the CPUs during the computation.
<b>Sys</b>	The cumulative time spent by all the CPUs during system-related tasks such as memory allocation.

According to the results, and after ran many times the code changing the sample file, I realized that the original Cat2.c file is faster than my program. This because the original functions are already factored and are smoother than my functions.