Instructions for Automated Testing

- Add your assembler program file in the Simple-Assembler Directory.
- Add the following line(s) to your **run** file in the **Simple-Assembler** directory of the **CO_A_P1** directory:

  I) Compile and run Java program:

  javac your_assembler.java

  java your_assembler

  II) Compile and run C++ program:

  g++ -o <output_file_name> your_assembler.cpp
  ./<output_file_name>

  III) Run python program:

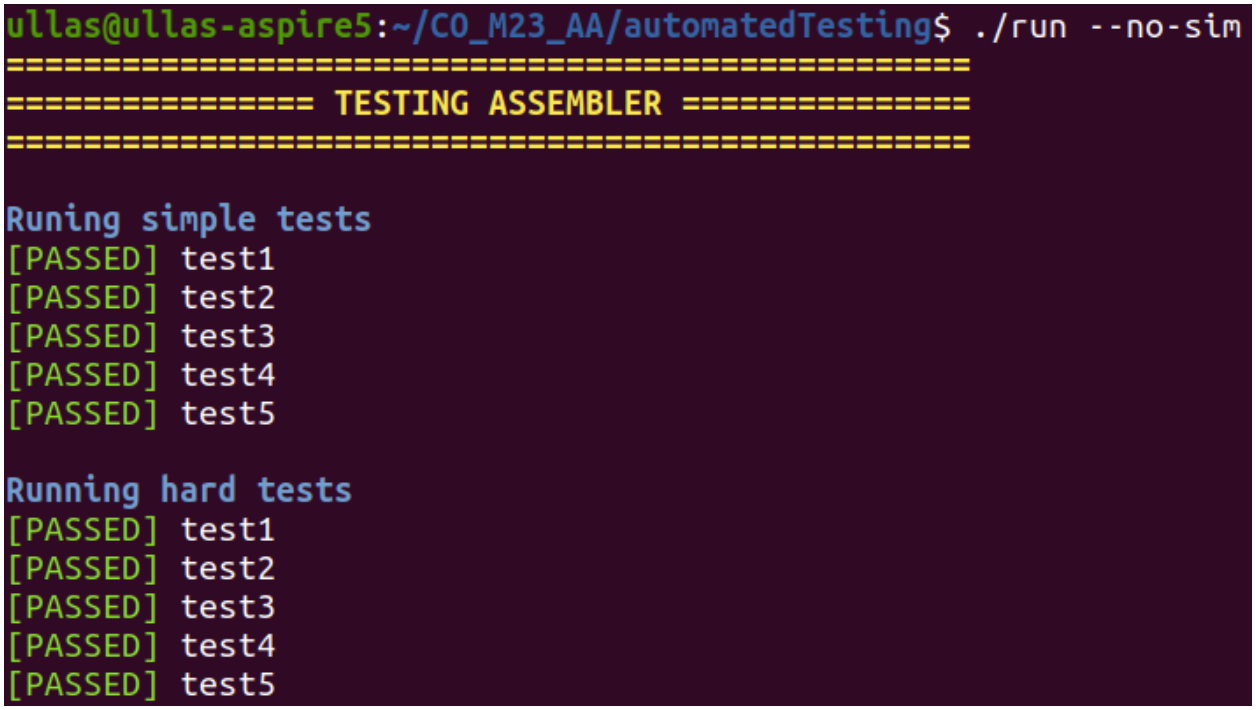
  python3 your_assembler.py

- Then go to the **automatedTesting** directory in your terminal and execute **./run** command (make sure your run file is made an executable file before running the run file)
- Also, make sure to run it with the **--no-sim** argument.

```
ullas@ullas-aspire5:~$ cd CO_M21_Assignment-main/
ullas@ullas-aspire5:~/CO_M21_Assignment-main$ ls
Assignment.pdf  automatedTesting  README.md  Simple-Assembler  SimpleSimulator
ullas@ullas-aspire5:~/CO_M21_Assignment-main$ cd automatedTesting/
ullas@ullas-aspire5:~/CO_M21_Assignment-main/automatedTesting$ ./run --no-sim
=================================================
================ TESTING ASSEMBLER ==============
=================================================

Runing simple tests
[FAILED] test01
[FAILED] test02
[FAILED] test03
[FAILED] test04
[FAILED] test05
```

The printed output will look somewhat like this (if your assembler program failed the test cases)



The printed output will look somewhat like this (if your assembler program passed the test cases)

- The errorGen files are run automatically, but they don't have a correct solution to compare with. Make sure your program prints any one valid error statement for the erroneous files and that you handle all possible corner cases for errors.

All the best!