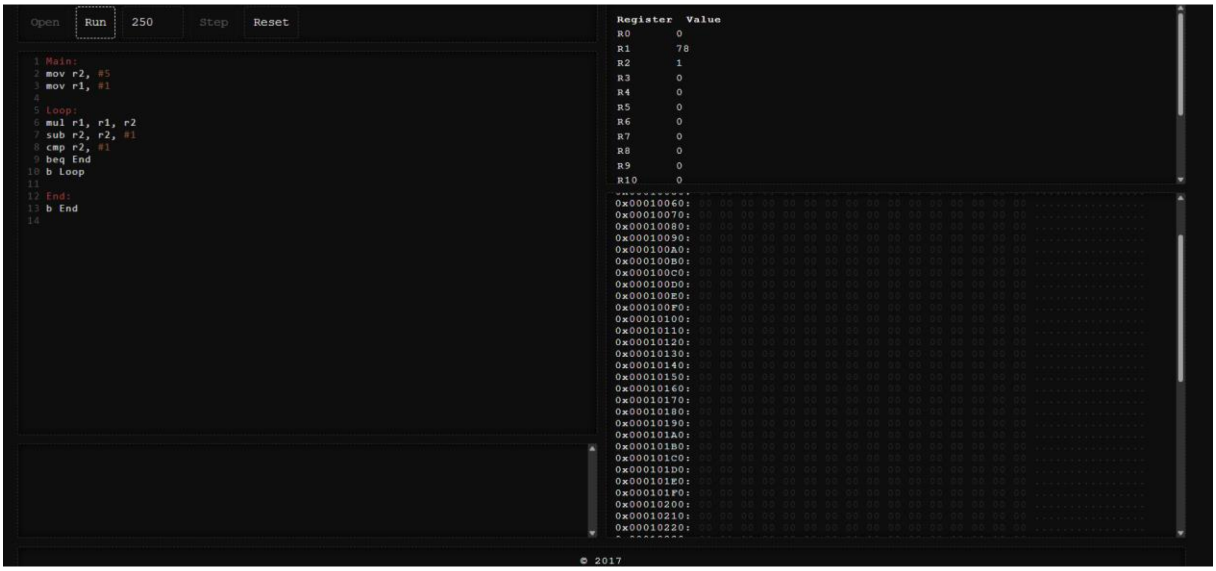
Template Week 4 – Software

Student number: 589531

Assignment 4.1: ARM assembly

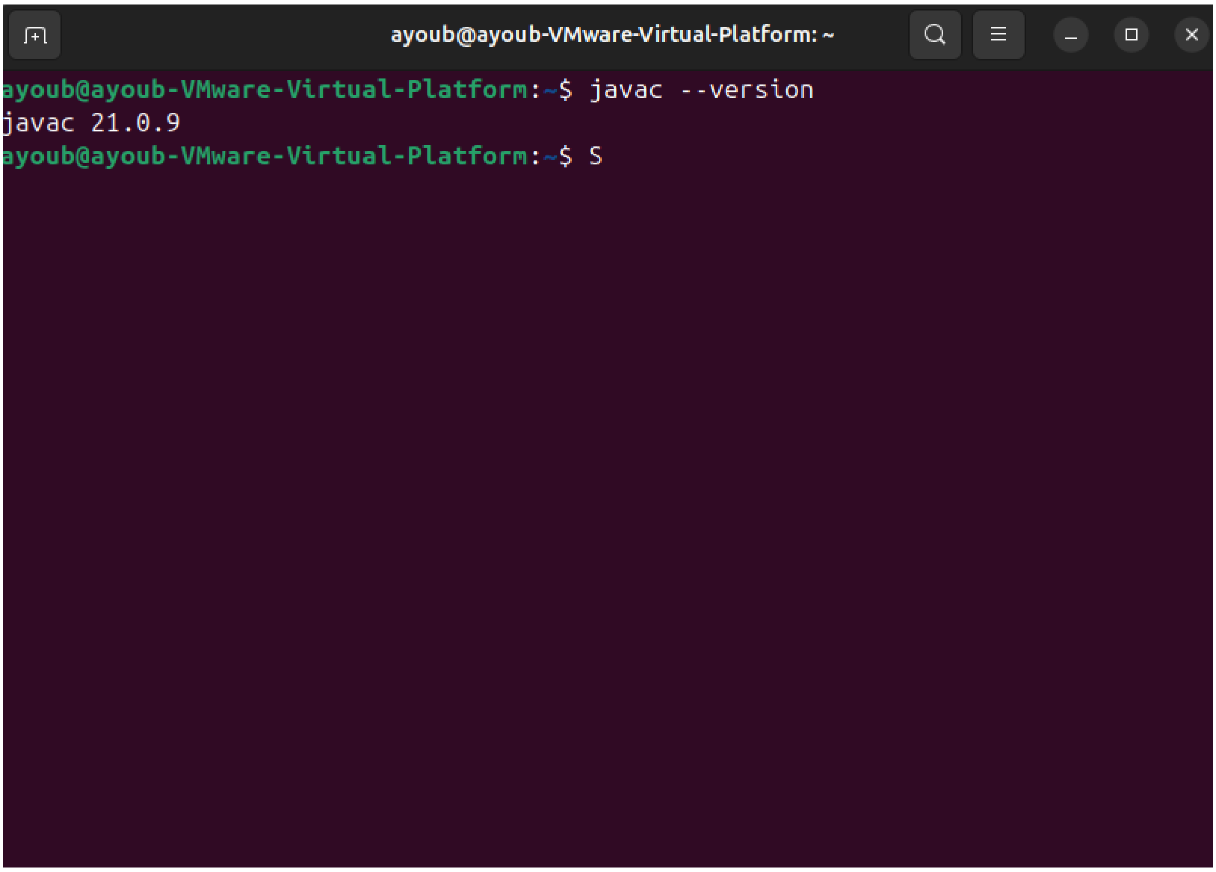
Screenshot of working assembly code of factorial calculation:



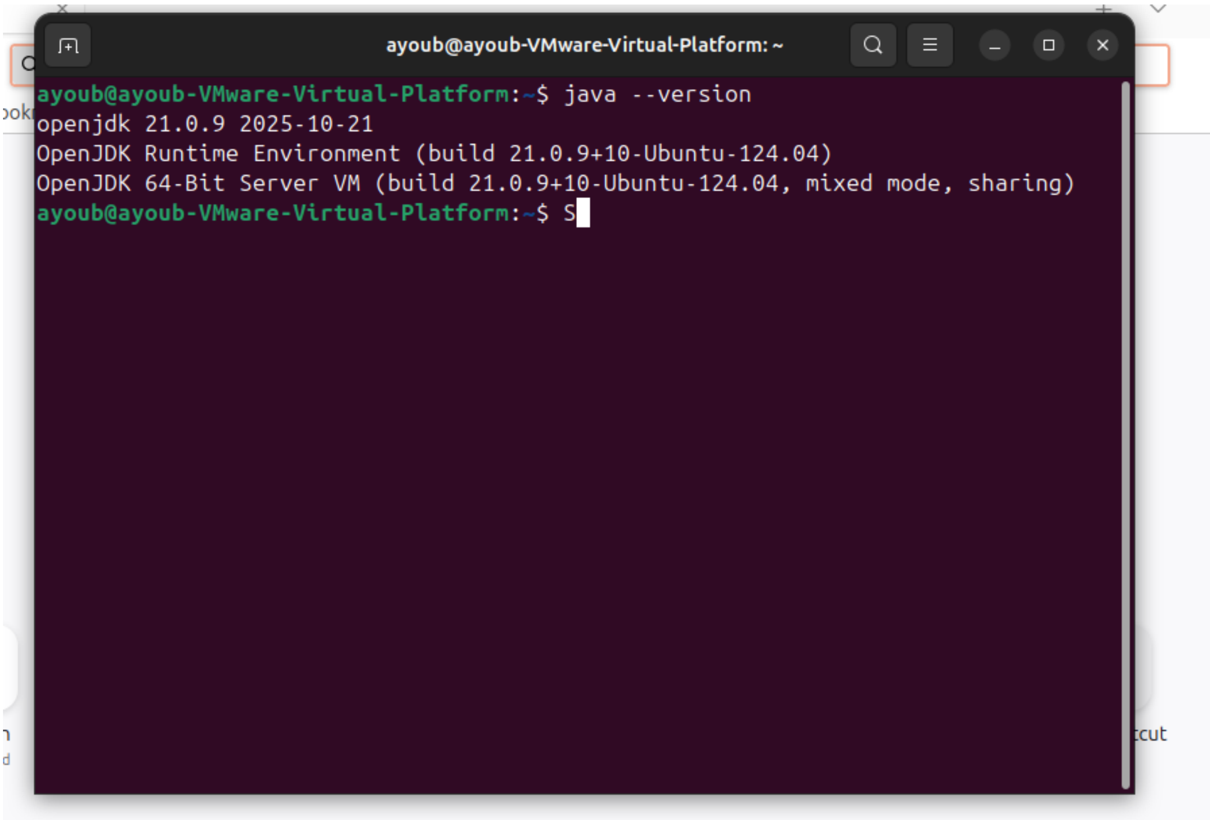
Assignment 4.2: Programming languages

Take screenshots that the following commands work:

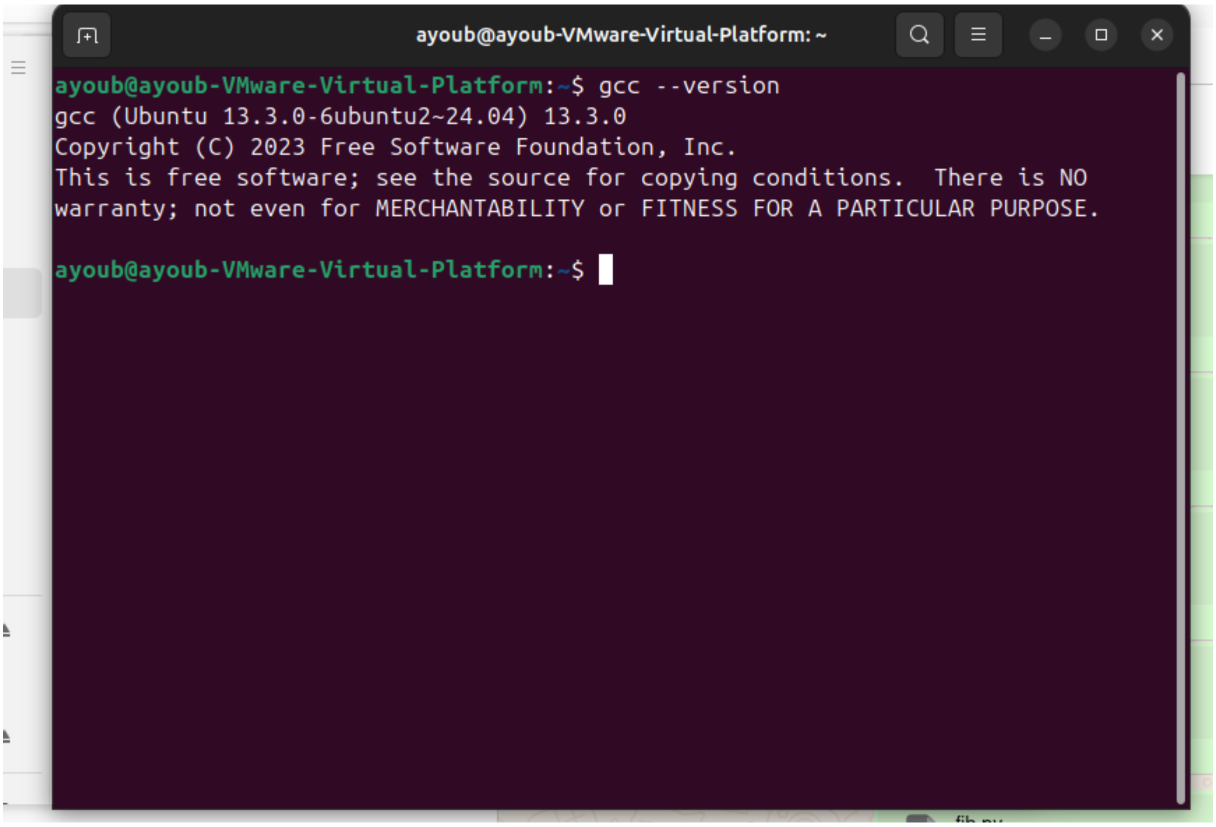
javac –version



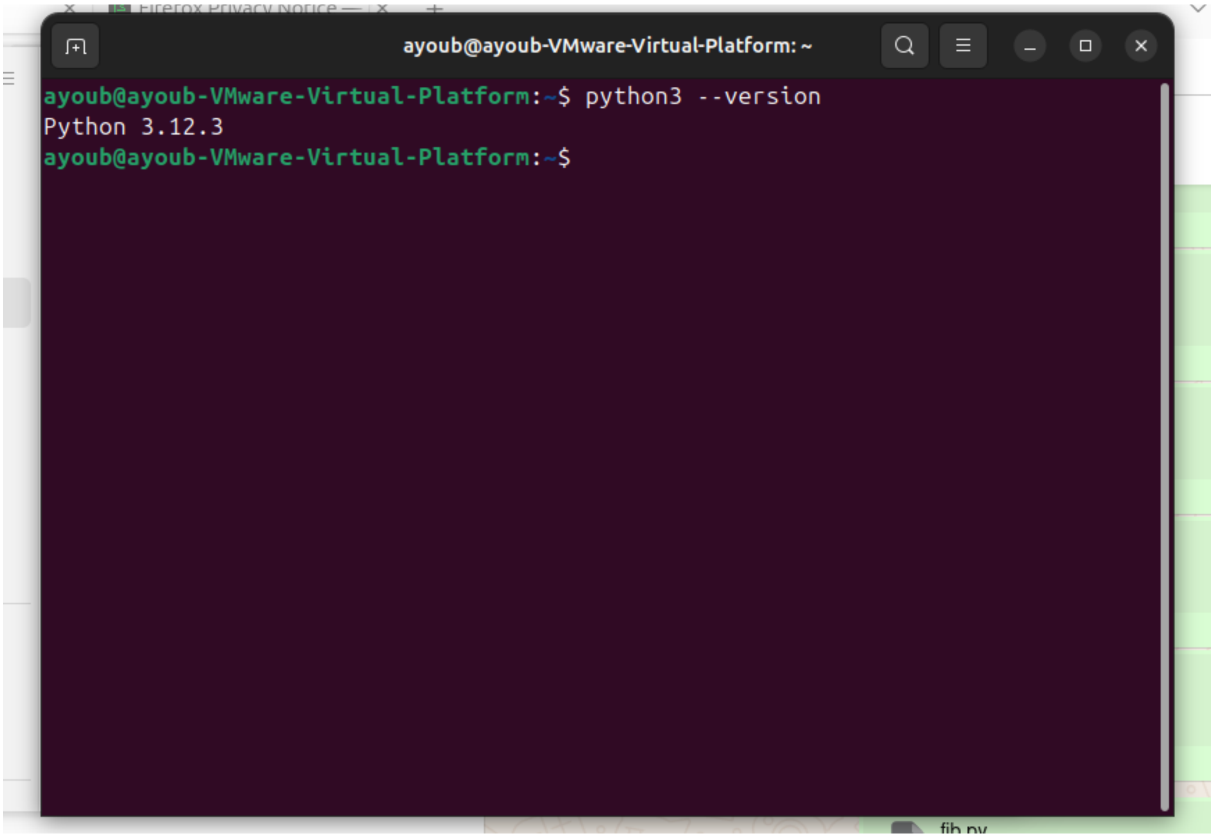
java –version


ayoub@ayoub-VMware-Virtual-Platform: ~
ayoub@ayoub-VMware-Virtual-Platform:~$ java --version
openjdk 21.0.9 2025-10-21
OpenJDK Runtime Environment (build 21.0.9+10-Ubuntu-124.04)
OpenJDK 64-Bit Server VM (build 21.0.9+10-Ubuntu-124.04, mixed mode, sharing)
ayoub@ayoub-VMware-Virtual-Platform:~$ S

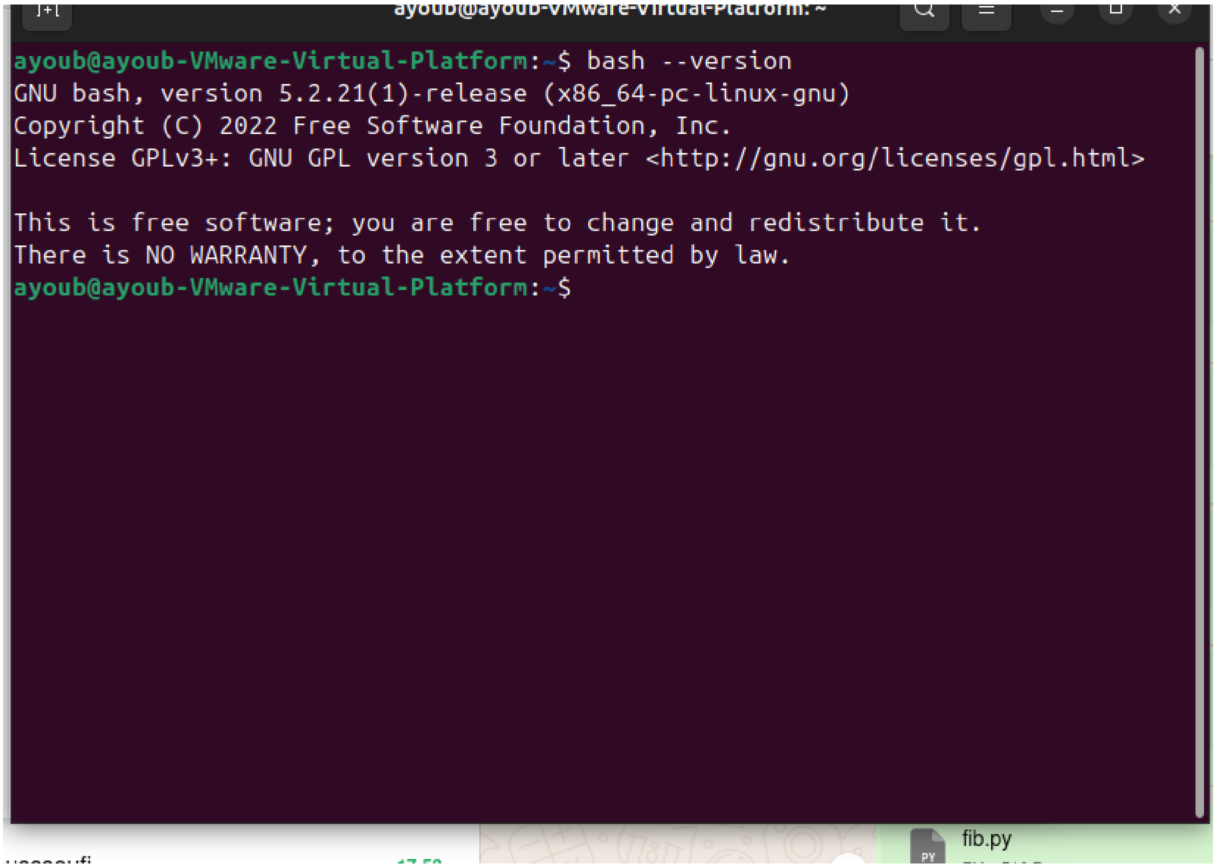
gcc –version


ayoub@ayoub-VMware-Virtual-Platform: ~
ayoub@ayoub-VMware-Virtual-Platform:~$ gcc --version
gcc (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
ayoub@ayoub-VMware-Virtual-Platform:~$

python3 –version



bash –version



Assignment 4.3: Compile

Which of the above files need to be compiled before you can run them?

Fib.c en Fibonacci.java

Which source code files are compiled into machine code and then directly executable by a processor?

Fib.c, nadat die gecompiled is dan word die leesbaar voor de CPU

Which source code files are compiled to byte code?

Fibonacci.java

Which source code files are interpreted by an interpreter?

Fib.py (python interpreter) en Fib.sh (bash interpreter)

These source code files will perform the same calculation after compilation/interpretation. Which one is expected to do the calculation the fastest?

Fib.c want hij heeft geen interpreter nodig om te runnen.

How do I run a Java program?

Eerst het programma compilen door javac Fibonacci.java, dat creeert Fibonacci.class en dat kun je runnen.

How do I run a Python program?

Die kun je gelijk runnen zonder te compilen.

How do I run a C program?

Door de file te compilen met GCC en dat maakt een executable file.

How do I run a Bash script?

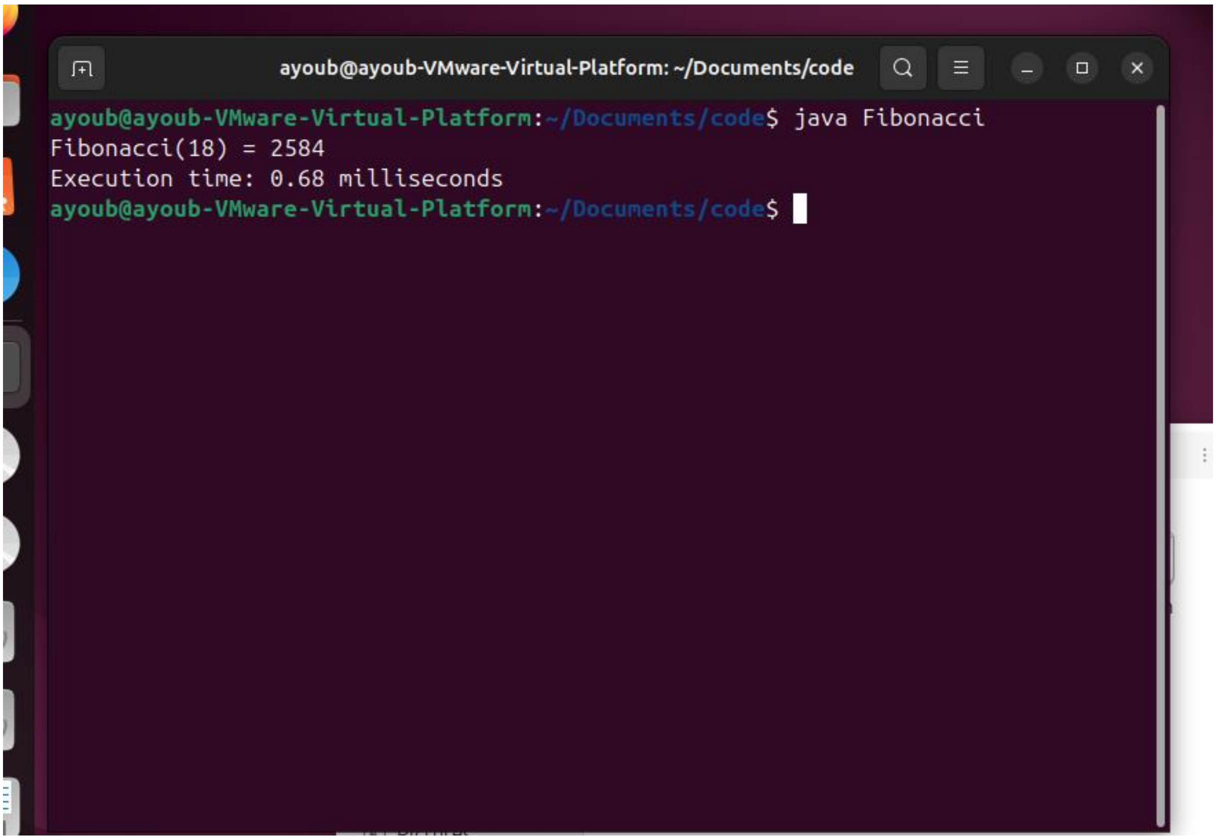
Compilen met chmod en daarna gewoon runnen.

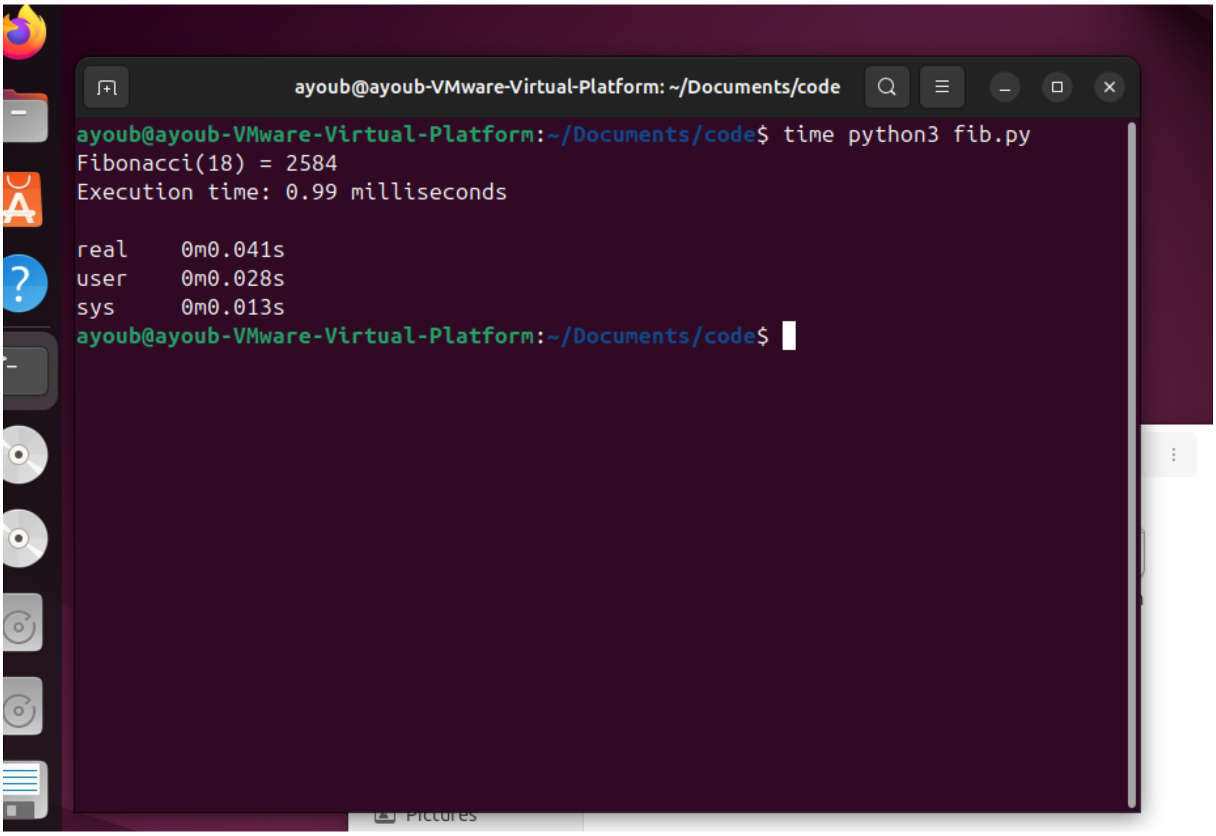
If I compile the above source code, will a new file be created? If so, which file?

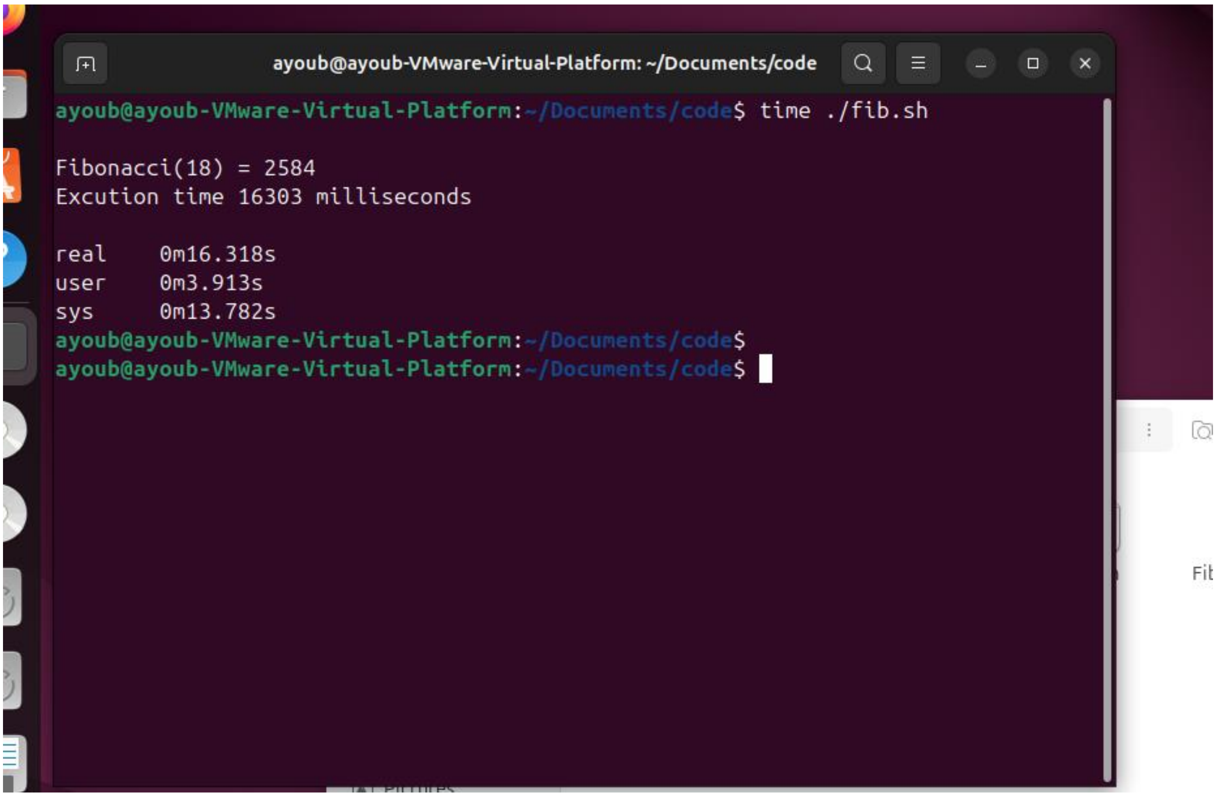
Ja, voor C komt een exe file, voor Java komt er een .class file dat je kan runnen, voor python en bash niet.

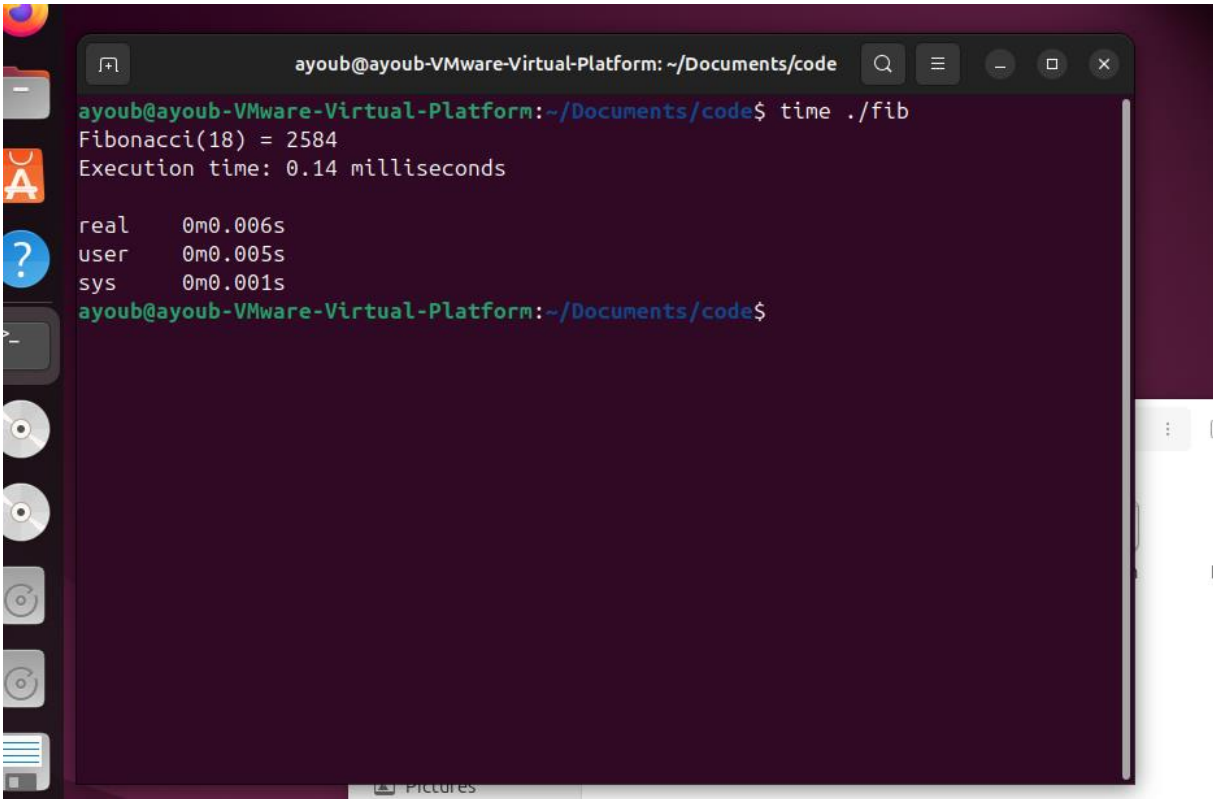
Take relevant screenshots of the following commands:

- Compile the source files where necessary
- Make them executable
- Run them
- Which (compiled) source code file performs the calculation the fastest?
  De snelste bleek C te zijn met sys 0.001s echt snel snel


ayoub@ayoub-VMware-Virtual-Platform: ~/Documents/code
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ java Fibonacci
Fibonacci(18) = 2584
Execution time: 0.68 milliseconds
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$


ayoub@ayoub-VMware-Virtual-Platform: ~/Documents/code
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ time python3 fib.py
Fibonacci(18) = 2584
Execution time: 0.99 milliseconds
real 0m0.041s
user 0m0.028s
sys 0m0.013s
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$


ayoub@ayoub-VMware-Virtual-Platform: ~/Documents/code
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ time ./fib.sh
Fibonacci(18) = 2584
Excution time 16303 milliseconds
real 0m16.318s
user 0m3.913s
sys 0m13.782s
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$

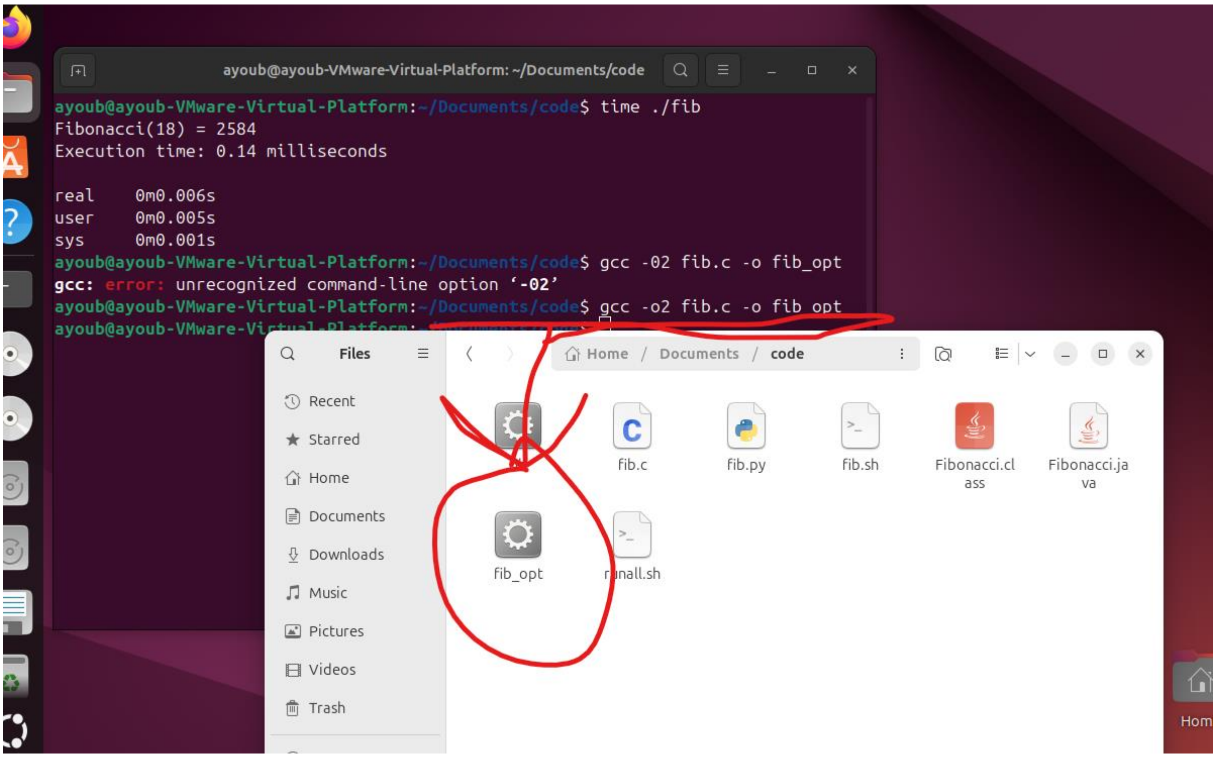

ayoub@ayoub-VMware-Virtual-Platform: ~/Documents/code
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ time ./fib
Fibonacci(18) = 2584
Execution time: 0.14 milliseconds
real 0m0.006s
user 0m0.005s
sys 0m0.001s
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$

Assignment 4.4: Optimize

Take relevant screenshots of the following commands:

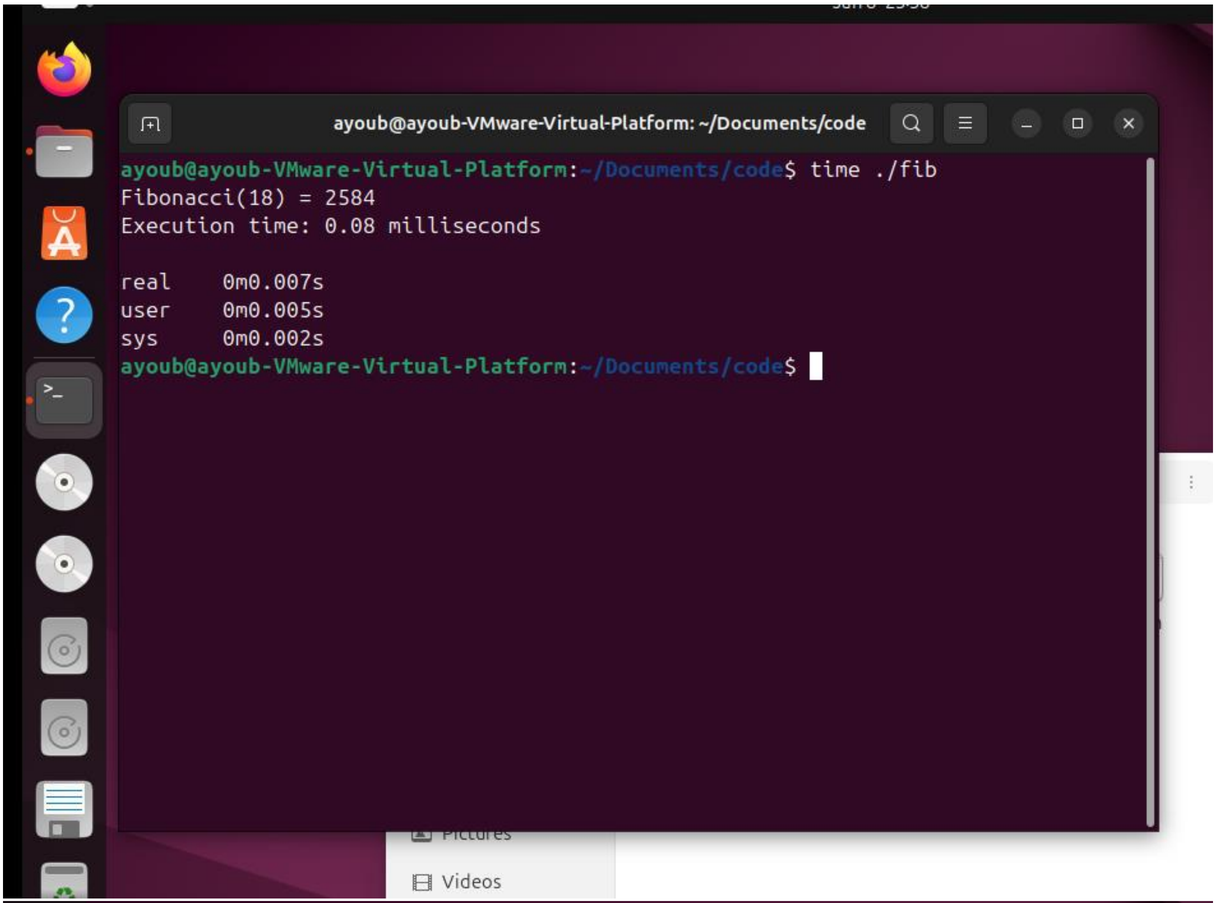
a) Figure out which parameters you need to pass to **the gcc** compiler so that the compiler performs a number of optimizations that will ensure that the compiled source code will run faster. **Tip!** The parameters are usually a letter followed by a number. Also read **page 191** of your book, but find a better optimization in the man pages. Please note that Linux is case sensitive.

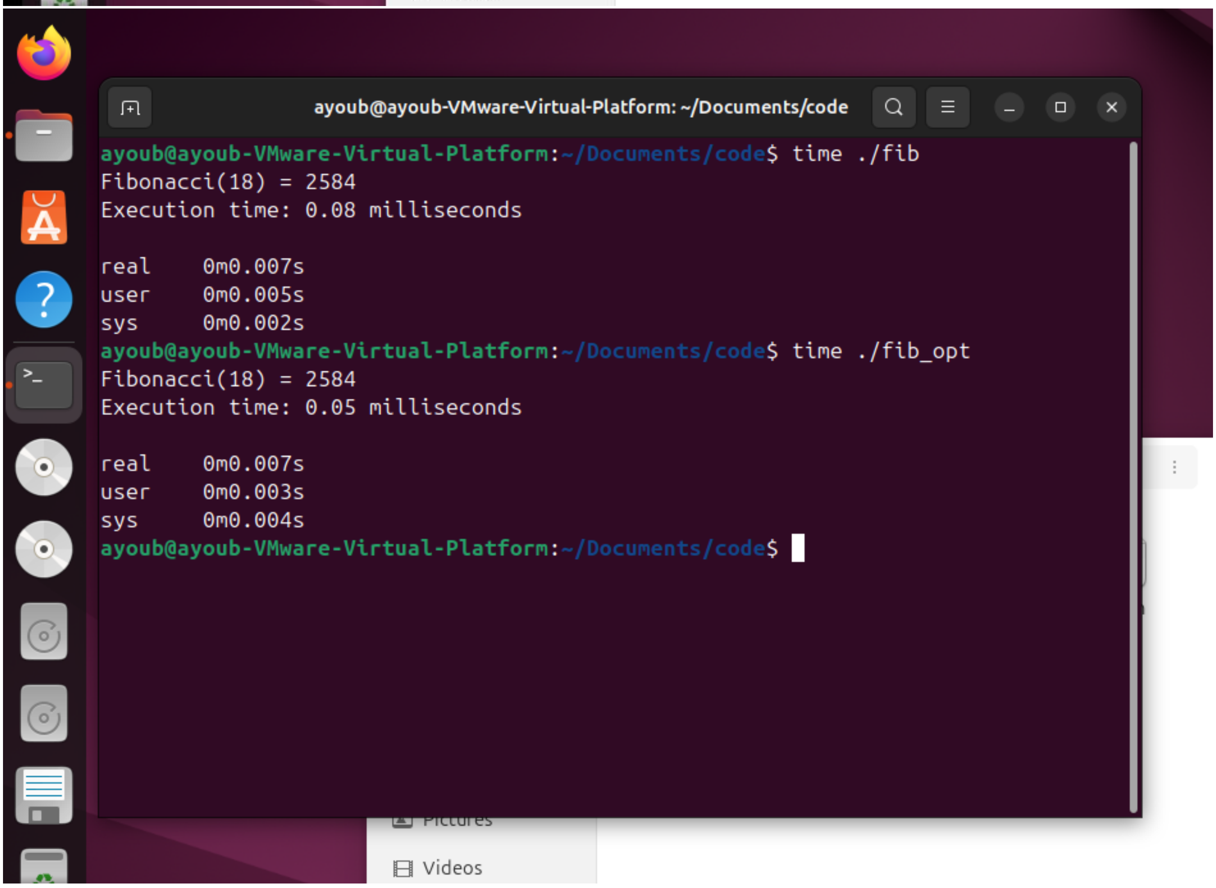
b) Compile **fib.c** again with the optimization parameters



c)

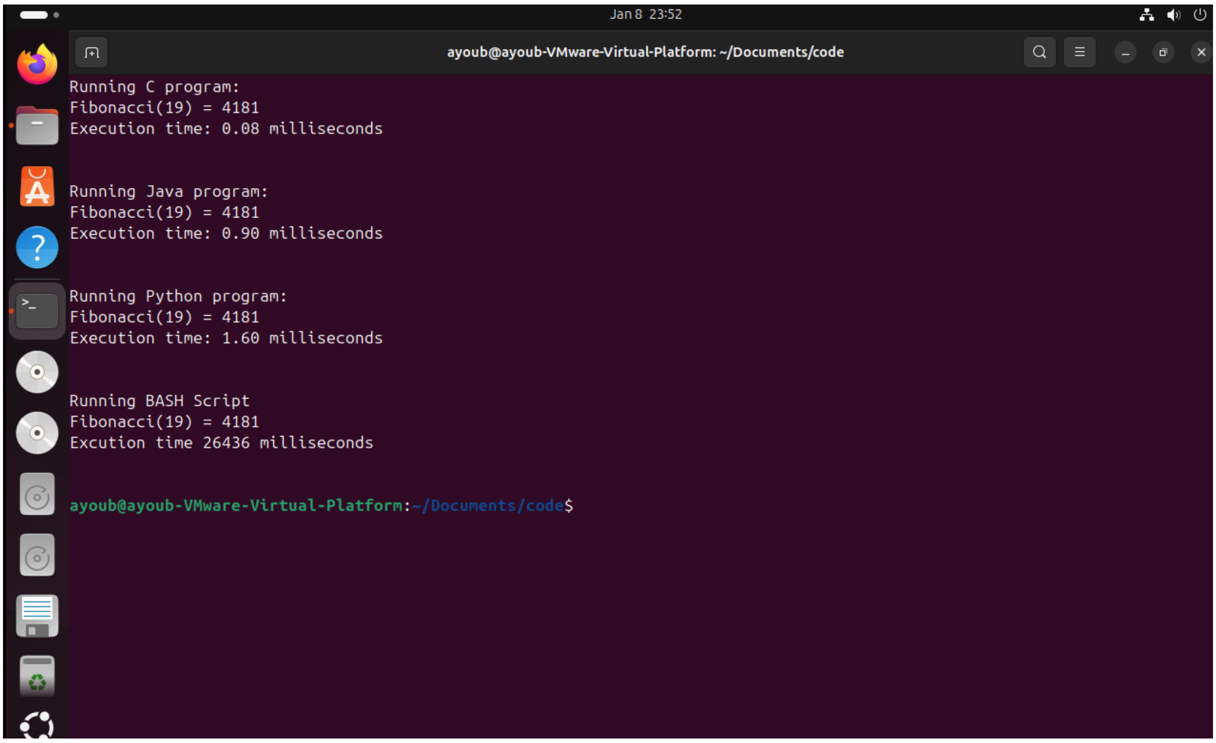
d) Run the newly compiled program. Is it true that it now performs the calculation faster?


ayoub@ayoub-VMware-Virtual-Platform: ~/Documents/code
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ time ./fib
Fibonacci(18) = 2584
Execution time: 0.08 milliseconds
real 0m0.007s
user 0m0.005s
sys 0m0.002s
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$
Videos


ayoub@ayoub-VMware-Virtual-Platform: ~/Documents/code
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ time ./fib
Fibonacci(18) = 2584
Execution time: 0.08 milliseconds
real 0m0.007s
user 0m0.005s
sys 0m0.002s
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$ time ./fib_opt
Fibonacci(18) = 2584
Execution time: 0.05 milliseconds
real 0m0.007s
user 0m0.003s
sys 0m0.004s
ayoub@ayoub-VMware-Virtual-Platform:~/Documents/code$
Videos

Zoals je kan zien runt die intotaal maar 0.03 miliseconden sneller dus inprincipe is die wel sneller ja.

e) Edit the file **runall.sh**, so you can perform all four calculations in a row using this Bash script. So the (compiled/interpreted) C, Java, Python and Bash versions of Fibonacci one after the other.



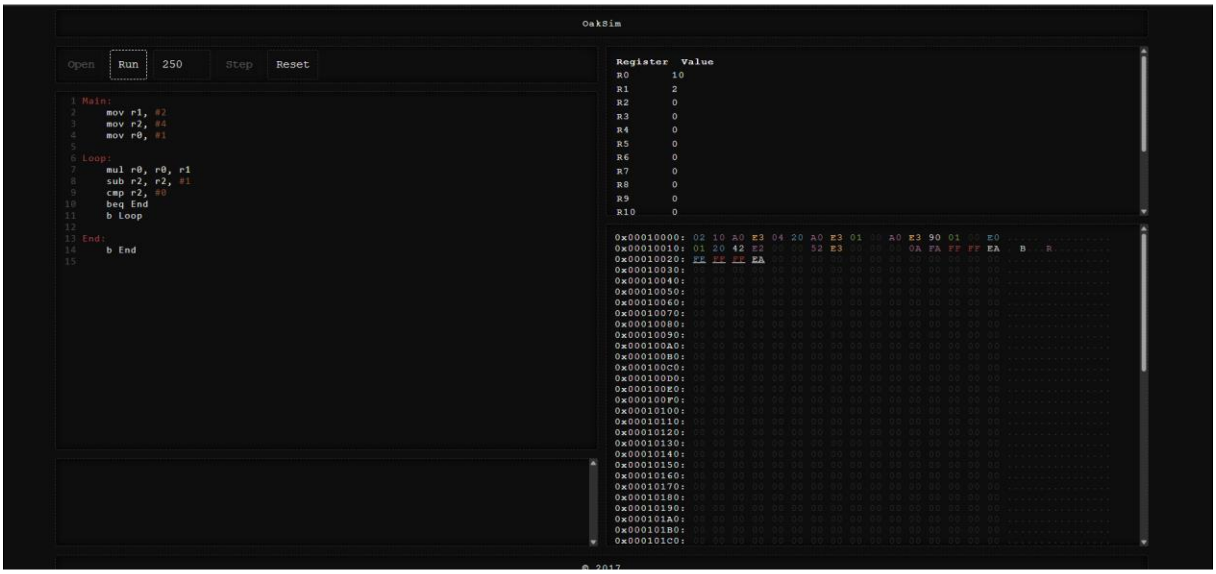
Assignment 4.5: More ARM Assembly

Like the factorial example, you can also implement the calculation of a power of 2 in assembly. For example you want to calculate $2^4 = 16$. Use iteration to calculate the result. Store the result in r0.

```
Main:

mov r1, #2

mov r2, #4


Loop:


End:
```

Complete the code. See the PowerPoint slides of week 4.

Screenshot of the completed code here.



Ready? Save this file and export it as a pdf file with the name: **week4.pdf**