Individual: ARM Assembly Language

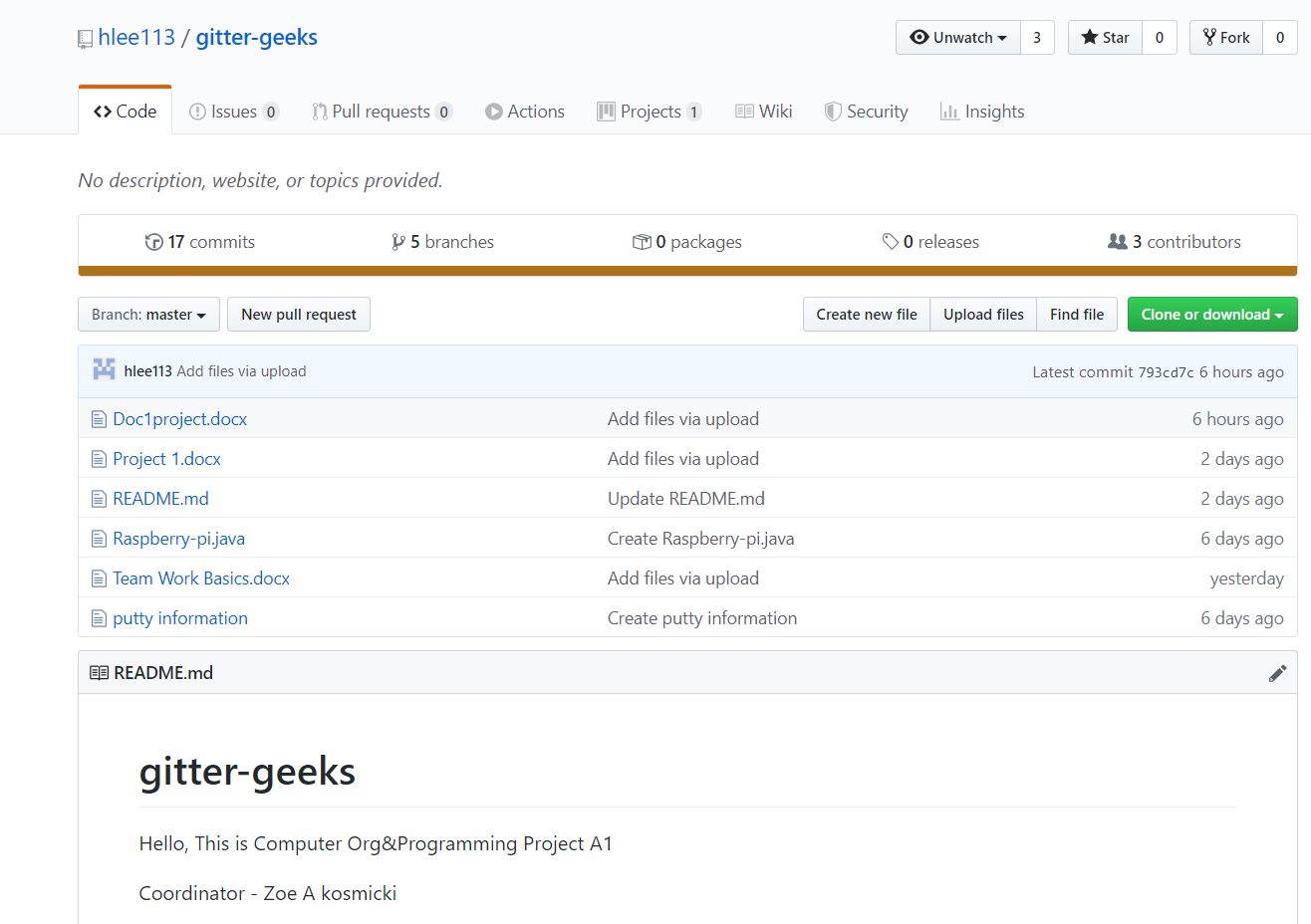
Tek Acharya

2/6/2020

TASK4. Raspberry Pi Installation and ARM Assembly Programming

A)

After creating a common account and a repository, we the gitter-geeks team worked together learning about ARM Assembly Language.



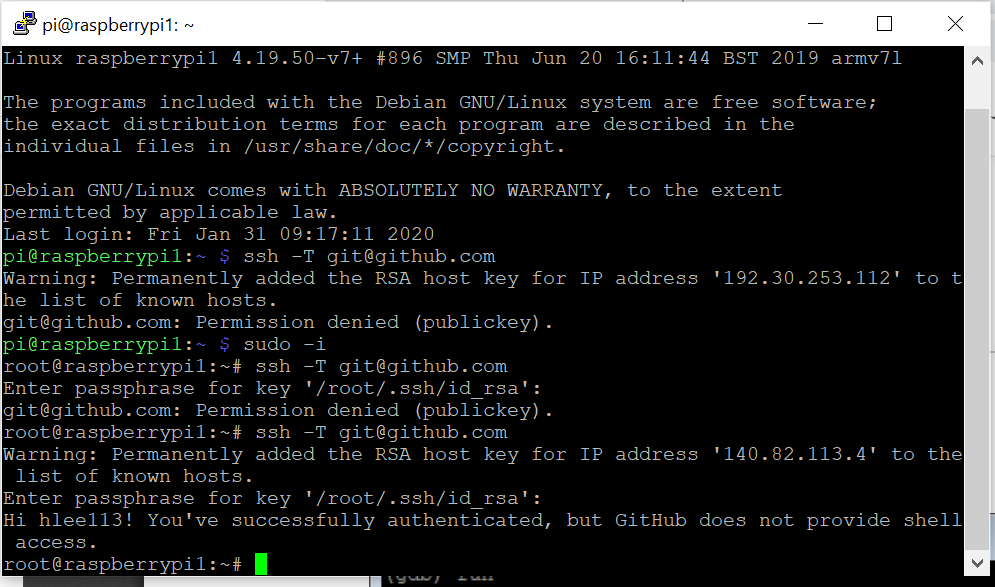
Pic: Our account with Repository named “gitter-geeks”

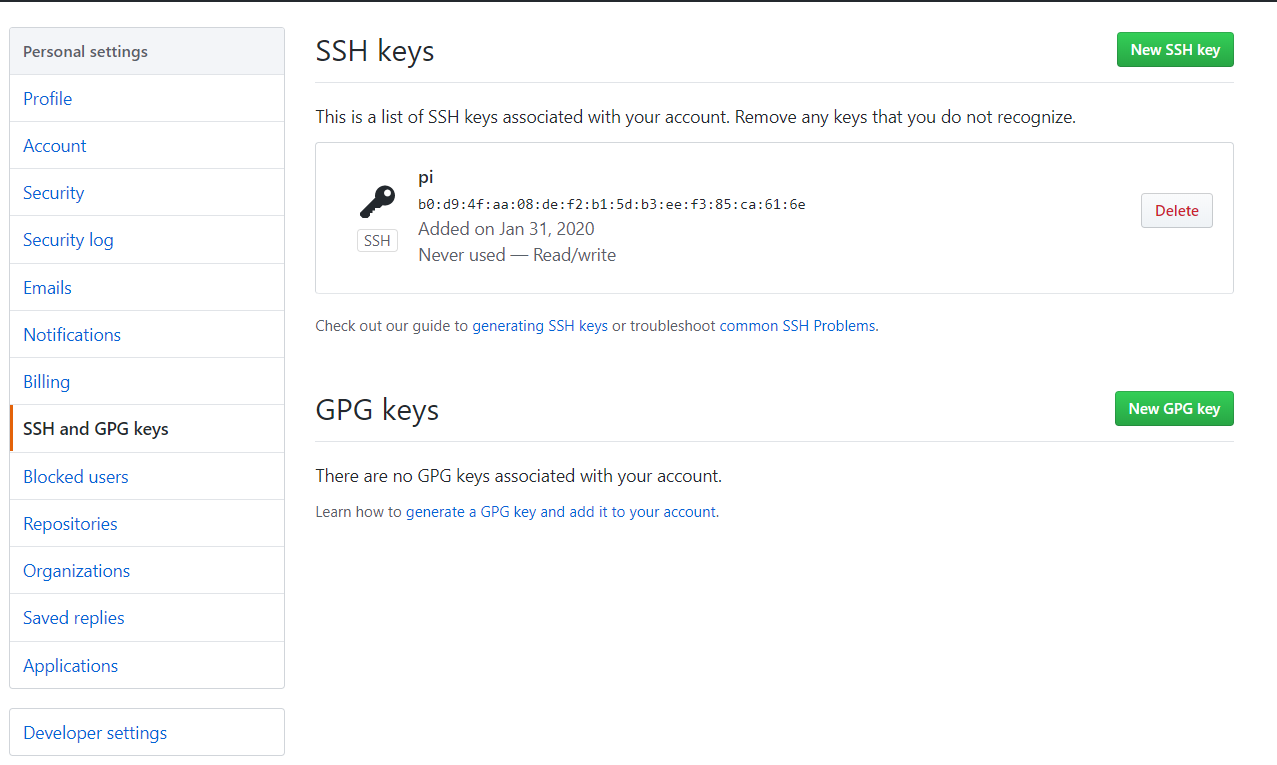
We, then connected our raspberry pi with the GitHub using SSH connection

A screenshot of a cell phone

Description automatically generated

We successfully installed and connected our raspberry pi using “pi account” via putty





We also tested the SSH connection under our repository by making test.git into our home/pi directory.

A screenshot of a social media post

Description automatically generated

The new folders and executable were created using commands given in the project.

B)

A screenshot of a cell phone

Description automatically generated

We compile, assemble and link first.s assembly program6) Did you see any output?

No

Why?

We don’t have any output to display. May be something going on in the register and memory.

A screenshot of a cell phone

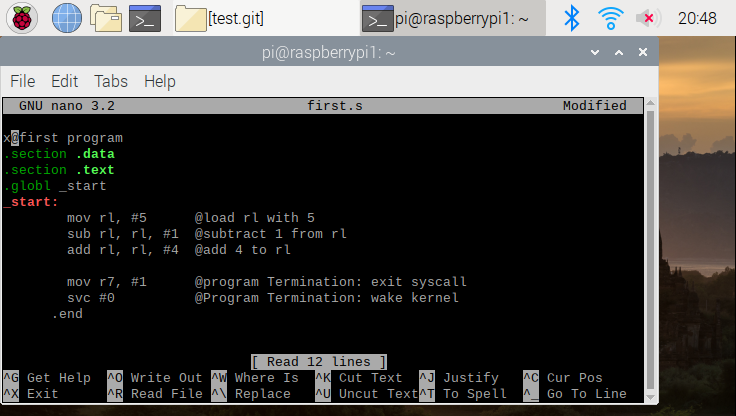
Description automatically generated

A screenshot of a computer

Description automatically generated

7) Debugging

Using GNU debugger we run our program as **gdb first** and use **(gdb) list** command to display our first.s program and started debugging. We set breakpoint at 11using g 11 command and run the code to see what happens into the registers and found the following output.



A screenshot of a cell phone

Description automatically generated

We get the output into the register r1 as expected as we first move 5 to register r1 subtracted 1 and added 4 resulting to 8 as shown.

PART II

Using the command from the above program, we have compiled, assemble, linked, run and debugged the following ARM Assembly program.

A = (A+B) – (C\*D)

Here: A = 10, B = 11, C = 7, and D = 2.

We named our new program as arithmatic1.s

A screenshot of a cell phone

Description automatically generated

Assembled, linked, and Debugging done

A screenshot of a social media post

Description automatically generated

Executable created

A screenshot of a cell phone screen with text

Description automatically generated

Nano Arithmatic1.s compiled

A screenshot of a computer screen

Description automatically generated

listed

A screenshot of a computer

Description automatically generated

Executed result

Here we got what we expected in the registers once we run and get the registers info.

The result of register2 (r2) was to be 11. r3 is 7 and r4 is 2 as expected. Here r5 has its value 21- the result of A added to B (qo+11). r6 stored the value 14 which resulted due to C\*D (7\*2). 11 stored the final value of A which is 7 which is the expected result of the given condition.

THE END OF THE REPORT