Developing Soft and Parallel Programming Skills Using Project-Based Learning

Spring 2020

gitter-geeks

Zoe Kosmicki

Vedansi Parsana

Michael Knight

Hyoungjun Lee

Tek Acharya

Planning and Scheduling

Planning and scheduling was difficult to start with, for as a team we all had various schedule differences and times we were unavailable. We all communicated using Slack and a more informal GroupMe, but because of our scheduling differences it sometimes took quite a while to do small tasks because we weren't around or available on GroupMe. Once we found a day that worked for us however, we got a lot done over three hours of working in a shared space. This was a very valuable experience for all of us; learning how to work together and compromise with our schedules meant that we very quickly learnt how to be efficient with the little time we had together. Moving forward in our project, we want to find easier and clearer ways to communicate and plan as a group so we can become more and more efficient in our work ethic, and get our tasks done in a timely and organized manner. Please see the Appendix to view our group's Scheduling and Planning timetable.

Teamwork Basics

Zoe Kosmicki

What is there to do to get the task accomplished and have the team members' satisfaction high? We said that we all needed to work on our communication skills, so we could all effectively update each other on where we were in our portion of the project. This would both help us get our tasks done quicker, if any of us had dependencies on other members, and keep everyone up to speed so we were all reassured that nothing had been rushed through or skipped over.

Work Norms: Work was a bit strange to assign this time around, since many parts of the tasks had already been started on/done before we had formally assigned work to everyone. We assigned tasks based on individual availability from our initial meeting until the due date, and honoring specific requests for certain portions of the project. We all decided on setting our deadlines at least three to four days before the official due dates, so we had time to look over everything together and fix any mistakes/misunderstandings before turning in our work. By setting this deadline, we could be sure that no matter what the work ethic of each individual in our group was, we could get everything finished and individually quality checked before the due date.

Facilitator Norms: The first facilitator was chosen based off of time available during the first project. We decided that we will rotate the position for every new project. The responsibilities of the facilitator is to make sure everyone is on the same page, communicating effectively, and setting short and long-term goals for this project.

Communication Norms: While we have a group Slack, we decided it's easiest and most accessible for us as a group to use GroupMe, since we all have easy access to our smartphones.

Handling Difficult Behavior: If a member happens to be overly talkative, to the point where they are talking over others, then I think it's a good idea to pull them aside to a one-on-one, and let them know that they're talking over others and being inconsiderate. We're a team and everyone's voices should be heard and taken into consideration. In a similar vein, if a member happens to be very combative and argumentative, I think that bringing them aside and asking them what the issue is is the best way. Getting them out of the group structure to really see what's wrong, whether they're uncomfortable, feeling unheard, or just having a hard time, will help them relax a bit more to help us all come to a solution.

When making decisions, if the team is having trouble reaching consensus, what should you do? I think that having everyone on the same page and working as a cohesive team is one of

the most important things to take away from this project. If this issue happens to arise, I believe that we should take a step back and try to diagnose the core of what our issue is, whether it's a misunderstanding, a wrong or differing answer, or another hard to visualize problem. Stepping back will help give us clarity, and can give us the insight needed to move forward all together.

What should you do if a person may reach a decision more quickly than others and pressure people to move on before it is a good idea to do so? Tell them that they're not being very considerate of the team, and ask them to wait until everyone is in agreement. There is no need to rush in this way, it will only lead to mistakes and miscommunications. It's all around a detriment to our finished product.

What happens if most people on the team want to get an "A" on the assignment, but another person decides that a "B" will be acceptable? I would still strive to do my best along with the other team members who want to get an A. I think an honest talk with the person who is okay with a B would be beneficial, so they could understand where we were coming from. I think as long as they didn't hold the rest of the group back and still communicated with everyone clearly, it wouldn't be a huge detriment. Of course, if they did start to hold us back, talking with everyone and assigning a fair grade would be in order.

Vedansi Parsana

What to do to get the task accomplished and the team members' satisfaction high? We should know every team member's ability, make everyone feel comfortable so that each person can open up, set basic rules for the team and coordination, use facilitator, and always be ready to help each member, to get the task accomplished and the team members satisfaction high.

How will work be distributed? The work will be distributed according to every member ability and switching tasks according to the team coordinator and every members approval.

Who will set deadlines? The team coordinator will set deadlines.

What happens if someone doesn't follow through on his/her commitment (for example, misses a deadline)? If someone does not follow through on his/her commitment, we will give definitely set a reminder before the due dates but even after that he/she forgets to submit the assigned task then we may assign that task to the other team member.

How will the work be reviewed? To review the work, we use slack, github and groupme. We meet every Friday to discuss what is completed and what is left so we can help each other if there are problems. Last but not the least team coordinator reviews everyone's tasks before the due date to make sure everything is perfect.

What happens if people have different opinions about the quality of the work? Firstly, every member of the group gets chance to share his/her opinion. And everyone listens to the ideas carefully and team coordinator decides about the final idea or opinion about the work.

What happens if people have different work habits (e.g., some people like to get assignments done right away; others work better with the pressure of a deadline)? People have different work habits, but there is a day or deadline of the group which our coordinator decides, we meet on Friday's as well to make sure everything is in pace and help everyone on the last meeting and make sure the work is done.

Will you use a facilitator? Yes, to make tasks easier to follow by the team, and make everyone aware of what they are doing.

How will the facilitator be chosen? Team coordinator will be the facilitator too.

Will you rotate the position? Yes, as the team coordinator will change for every assignment automatically the role for facilitator will be changed for every assignment.

What are the responsibilities of the facilitator? Responsibilities of a facilitator is to keep and ensure that everything is on track, and ensure that everyone is assigned the task properly. According to me, there are two types of people in most of the groups which are - mostly arguing or too quiet. A person who argues the most can be a person who either knows everything or is confident about what he/she is doing for that we should ask him patiently to listen to everyone's opinion and not aggressively fight for his arguing. The second is too quiet behavior which I think people may feel shy to speak in public or cannot mingle with people for them we should always make him/her comfortable and ask their opinion, appreciate his/her work.

When making decisions, If the team is having trouble reaching consensus, what should you do? If the team is having trouble reaching consensus, then the coordinator should listen to each members working ideas and decide what to do knowing all pros and cons.

What should you do if one person may reach a decision more quickly than others and pressure people to move on before it is a good idea to do so? We will ask him/her to listen to everyone's idea which is the basic rule for the team so that nobody feels bad afterwards.

What happens if most people on the team want to get an "A" on the assignment, but another person decides that a "B" will be acceptable? If most of the team want to get an "A" on the assignment, but another person might decide that a "B" will be acceptable, then we motivate that person to get an A which is never a bad opinion. But still that person decides no matter what he/she wants B than according to the majority we will decide an A and that person must receive partial grade for the specific project.

Michael Knight

What to do to get the task accomplished and the team members' satisfaction high? Come together to help ease the difficulty of the assignment for everyone in order to accomplish the task. If everyone is working towards the same goal then there would normally be low stress and result in a higher satisfaction on completing it.

Work Norms: Work will be distributed by the Team Coordinator. Team Coordinator Sets deadlines. If someone doesn't follow through with their commitment, the unfinished task/tasks will go to who volunteer or volunteers, or to whomever is available to pick up the slack. The work will be reviewed by everyone. If different people have different opinions of the quality of work. It should be brought to whom did the work first, if an agreement can't be made then it should go to the team coordinator's attention for a decision to proceed further on the matter. There should be deadlines for each step so if someone's work who is dependent on another's work to be finished won't be held up or have to wait too long.

Facilitator Norms: Yes, We'll use a facilitator. A facilitator is chosen by random or by volunteering. The position is rotated for each assignment. The responsibilities are assigning tasks, assigning deadlines, and settling disputes.

Communication Norms: Communication should take place in person and on slack, and constantly to make sure no one is falling behind. Meeting Norms: Everyone's schedule is different with classes and jobs. Everyone is responsible for setting aside time for meetings. Everyone has come to an agreement on when to hold meetings. The Library is where we have chosen to meet up. If the person or persons whom are tardy for a meeting and has let us know ahead of time. With reasonable time we'll wait before starting. If someone misses a meeting without a reasonable excuse, then it'll be noted. And they will be caught up outside of the meeting. If they miss several it will be mentioned to the professor.

Consideration Norms: If it is not a distraction food is allowed. Smoking is not allowed. If someone is dominating the discussion they will be asked to let someone speak their mind on the subject and wait their turn. Norms can be changed with everyone in agreement to accommodate someone who is uncomfortable with it.

In Handling difficult behavior we've chosen If someone is too quiet: We'd investigate with sensitivity in mind to find out why the participant is being too quiet. It could be from their personality as a shy person, not feeling well, to something as they are embarrassed from a disability and don't want to draw attention to it. After finding out the reason we could think of ways to handle it and help reassure that with their input we can better overcome the tasks ahead of us.

Made a complaint: all complaints can be addressed with who one is having a problem with. If a resolve can't be met, then the team coordinator should be brought in to handle the dispute. If both parties can't come to terms with the solution decided by the coordinator if it is detrimental to one or both parties and they feel that it still is going to affect their work, then could ask for guidance with the professor.

If most of the team wishes for an A and there is someone who is fine with getting a B: if the work done by the person who decides getting a B is acceptable isn't satisfactory of the rest of the group then it may be redone by 1 or more other people and their actions shall be recorded for the professor's judgement in grading.

Hyoungjun Lee

What to do get the task accomplished and the team member's satisfaction high? Everyone sticks closely to same page, each member gives feedback to each other, try to avoid conflicts, members rotate coordinator, and respect each other.

Work Norms: Work must distribute fairly and equally, get to know each other and know that who has strength of specific part and weakness. The coordinator should set up the deadlines, mostly planning feature. Work will be reviewed in GitHub and google docs, by online for every team member. If someone has different opinion about quality of works, first we need to meet up and communicate, if that does not work, divide tasks into more than we got, distribute work more fairly. This is teamwork, at least set up for group members working into progress, as we have about 2 weeks, we could manage by communicate.

Facilitator Norms: Yes, we assign members to each task, who can mainly focus and then help each other's parts if there are any problems. We can choose facilitator by communicate in our group, discuss about who want to do first, since we must rotate coordinator, someone should take spot. The main responsibility of facilitator is keeping the group progressing to the right direction, and make sure to get everyone's participation, e.g., we are on same page.

Communication Norms: Communicate should takes whenever group member has problem and talk about what we have done, and future planning, for our group, gitter-geek, we communicate in several ways not to have problem with communicating, we use slack, GroupMe, google docs and email.

Meeting Norms: One thing biggest problem is meeting up, because this is hard to make 5 people into same time, same place, for gitter-geeks, thankfully we could make a time on Friday lunch we book a library room which is good place to hold meeting. If people late showing up meeting, we can talk and discuss first and when he/she comes we can tell he/she what we have discussed. If a group member misses or misses several meeting, for 1 time, we can send an information through online, however, if he/she misses several meeting, we would think that he/she does not want to work together, we can put low grade for table chart.

Consideration Norms: If group member says "excuse me, I'm so starving, I will finish quickly" or "give me just 1 minute" even say that, it would distract group discuss, but as long as we are only with group member, could be excused. If someone is dominating discussion it will not be good discussion, should be share everyone's thought not the one person's thought

Too quiet: If one of member is too quiet during the discussion, do not judge that he/she does not know anything, could be shy characteristic, just give an extra effort to him/her asking question first that is there any problem or do you have any opinions to say?, then he/she will response and will get to in our conversation

Complaining: If one of member is complaining a lot, group members should listen first and figure out how should we manage the problem, and then persuade him/her with logical reason or meet the professor how we going to solve this problem

If the team is having trouble reaching consensus? In my opinion, most of time decisions will be made in consensus because our group members are odd number, which is 5, could not have drawn. If we can not figure it out ourselves, we have to talk to our wisdom professor

What should we do if person may reach a decision more quickly than others and pressure people to move on before it is a good idea to do so? Once we have more time to reach a decision, and decision that we made is not good idea, we could think more time and discuss about is there any good ideas? If there is not better idea after having discuss, we just go forward with that decision

What happens if most people on the team want to get an 'A' on the assignment, but another person decides that a 'B' will be acceptable? If one person does not want to get an A and not giving a fully effort on his/her task, we could ask him/her to help us out, if not, we have to cover his/her work and write down in report about less complete to instructor

Individual and Team Goals: Individual Goals should be having high level of programming knowledge, and know how to interact or how to work together with team. Team Goals should be everyone work as teamwork for each other, help each other, satisfy rubric's grade and get the best grade.

Tek Acharya

Learning Teamwork Basics. Tasks to accomplish to get a team member's satisfaction high: Know each other in the group and their weaknesses and strengths. Set up basis rules to be accomplished, implement and monitor it. Assign one of the members as a coordinator.Always stay communicated. Avoid conflicts and resolve common issues amongst the group members. Work on the assignment with collaboration and a sense of respect.

Work Norms: Work must be distributed equally and favorably. Keeping in mind the strengths and weaknesses of a person and despite other workloads such as assignments from other classes or having a full-time job. The coordinator with the help of the team members should set the deadlines. If someone does not follow through or misses a deadline, then the other team members must decide about the person's portion of workload that has already been

contributed. The rest of the team members must mention this portion under *the Work Breakdown Structure* of the section *Planning and Scheduling* section of the *Report Writing* of the project. The team members can repair the loss incurred by that person as much as they can. The work can be shared through a common platform such as GitHub and Google Docs and work together online. If people have a different opinion about the quality of work, first we will refer to Google or books, if that doesn't work, we can either vote among the other people or see the instructor. As we are given fifteen days to complete the task, we should have enough time to adjust between all the odd situations. However, regardless, people must submit the assigned task within the deadline. If not, the rest of the team members will assign only the portion of the contribution that was done on-time.

Facilitator Norms: Yes, we will use a facilitator; a different person in every assignment. A facilitator can be chosen by communicating within the group. If no one is willing to take responsibility; the previous group coordinator will assign one. Moreover, in our group, we have predefined our next facilitator. The main responsibility of the facilitator is to coordinate among the group members (sometimes outside the group members) in achieving the highest goal. Other than that, the group facilitator makes the group members respond to their tasks and reinforce the preset definitions to keep it moving and decision making. Also, the facilitator finds an alternative to a stalled situation.

Communication Norms: We must communicate to share the progress of member's tasks and overall progress of the group. We also need to communicate if someone has a problem. We, gitter-geeks, are communicating using slack and use text messaging if no response through slack. We are also sharing the Google Doc and editing tasks online.

Too Quiet: If we encounter anyone with this behavior, we will find out why the person is overly quiet by asking questions such as if the person is okay, asking some project-related questions, or opinion about certain decision making among the group. Once we receive the response, we will find out if the person is unwilling, lazy, tired, shy, unsure or unwell. We will then use our wisdom to best fit the person and try to motivate the person while looking for hidden strengths that can be useful. If any of these steps are helpless for us going forward, we will contact the instructor.

Made a Complaint: If a person made a complaint, we will carefully review if the complaint was legitimate. If the complaint is legitimate then we will find a way to correct it including the person who complained. If the complaint is illegitimate, we rest of the group members will find the best way to approach; this can sometimes be confidential communication or by re-discussing the material with the group and finding the actual solution keeping in mind the complaint but without treating it as a complaint.

While deciding, if we are unable to reach a consensus, we will do some research and use available yet reliable resources such as textbooks, Google, and Instructor. Sometimes it can also be a good idea to reach out to inter-groups members.

If a person reaches a decision too quickly and pressures the other member to follow with his/her wrong decision than the rest of the team members must reiterate the rule of thumb; if everyone agrees with the current decision. We can also reiterate the subject matter and discuss it to see if some other decisions could be a better one. If the problem persists then one of the other members will have to talk separately.

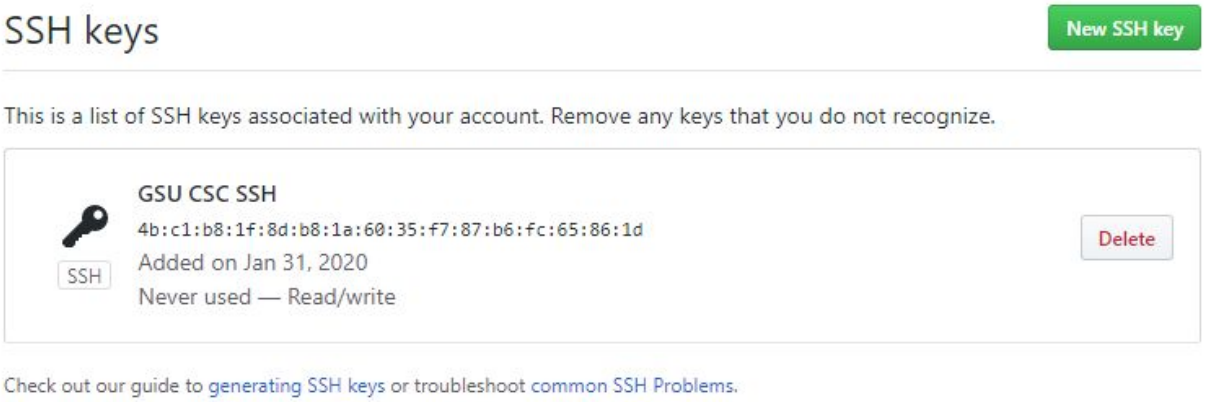
If most of the group members wanted an "A", while one of the members is fine with "B", then we will need to request the person to be helpful whenever needed even if it is less important

to him/her. If we didn't find him/her helpful or communicating; we will do our best to cover-up the loss for our better grade and impartially mention the person share in the *Work Breakdown Structure* and report the person to the instructor.

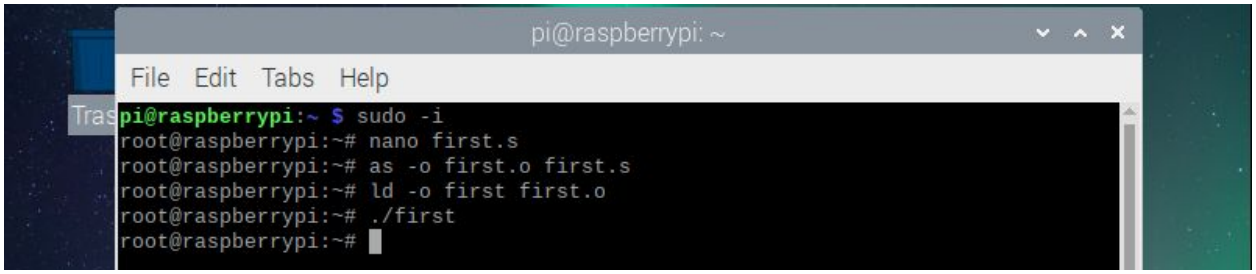
Raspberry Pi Installation and ARM Assembly Programming

Zoe Kosmicki

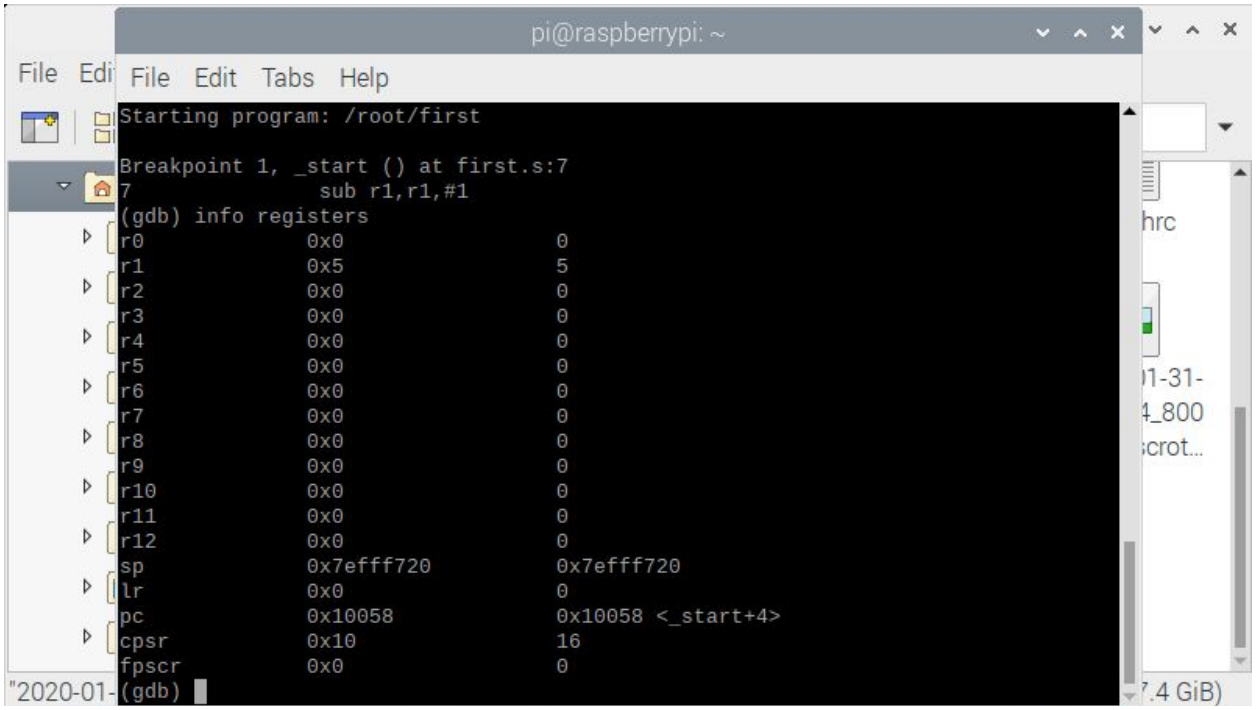
I connected my Raspberry Pi to our GitHub repository using a SSH key, and created a test file so I could push files from my Pi to GitHub.



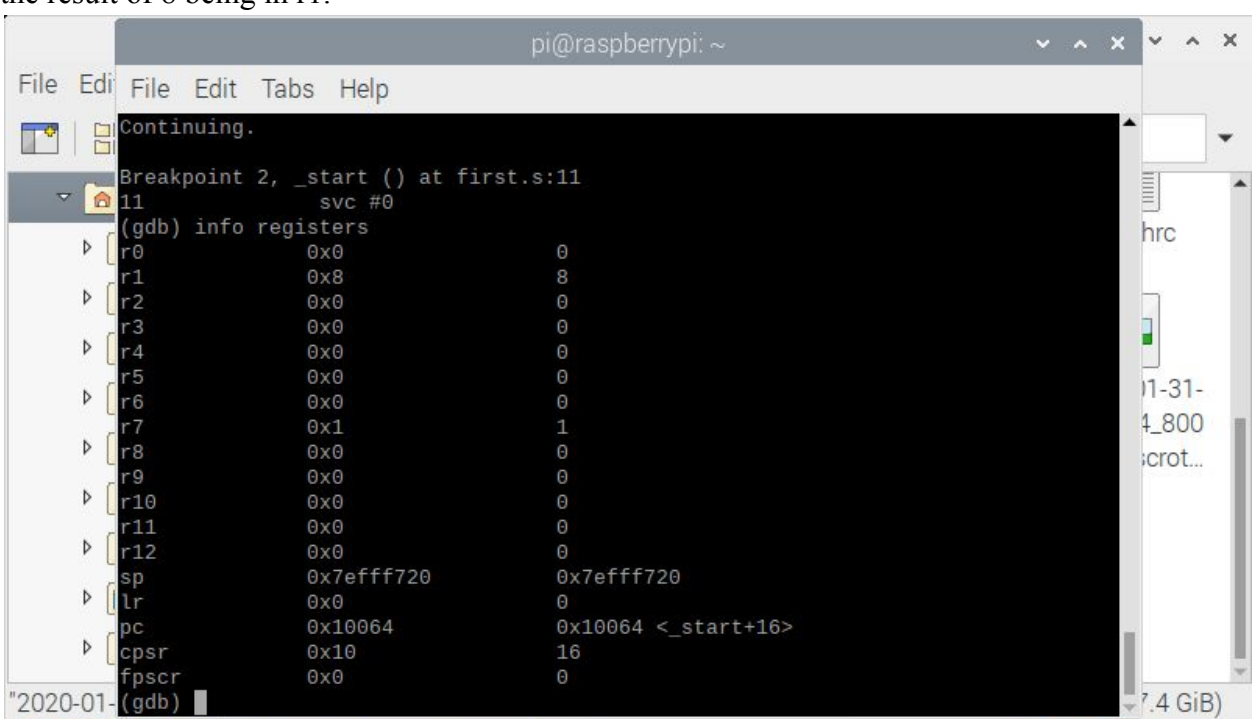
I created the first project using the nano editor, and ran it using the ./first command.



Of course, no output is shown because everything is done within the registers, no value is actually returned. In debug mode, you can see what is actually happening within the registers.

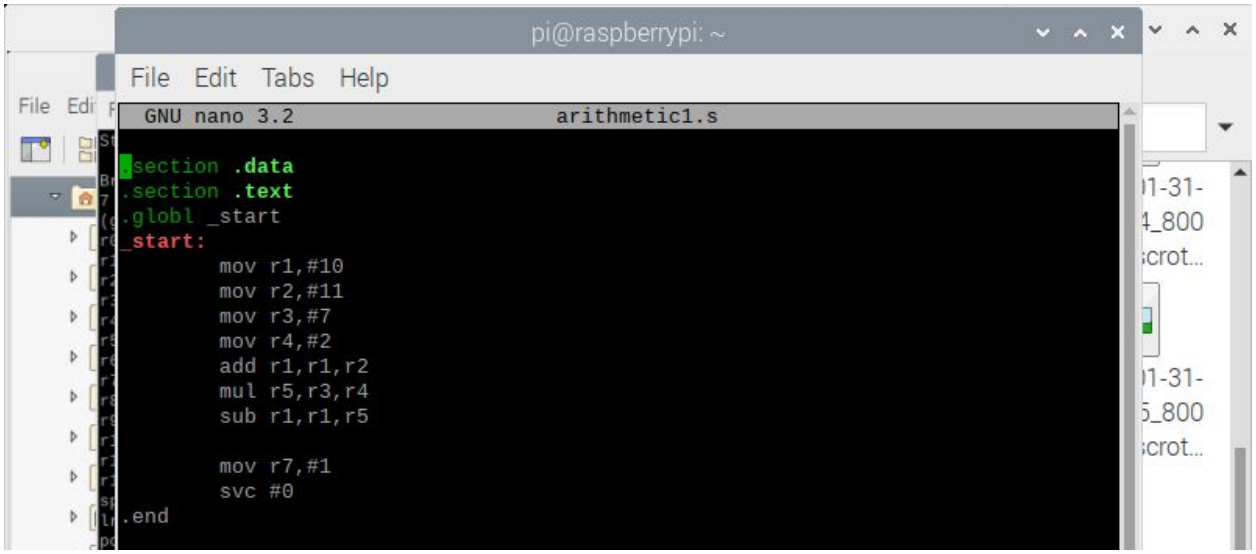


At the first breakpoint that was set, at line 7, we can see that r1 has been loaded with 5. Right after this, the value in r1 will be subtracted by 1, and then have 4 added to it. This should give us the result of 8 being in r1.

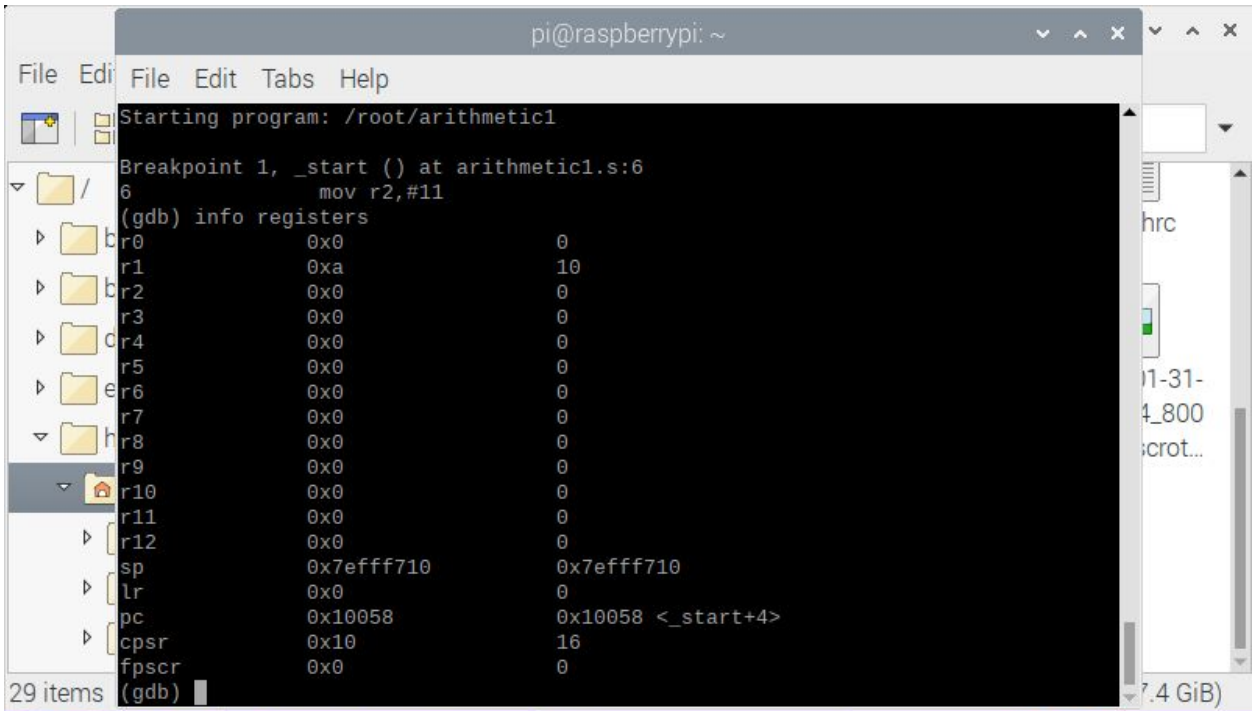


As expected, in r1 after the above instructions, 8 is in r1 and r7 has been loaded with 1 to start exiting syscall.

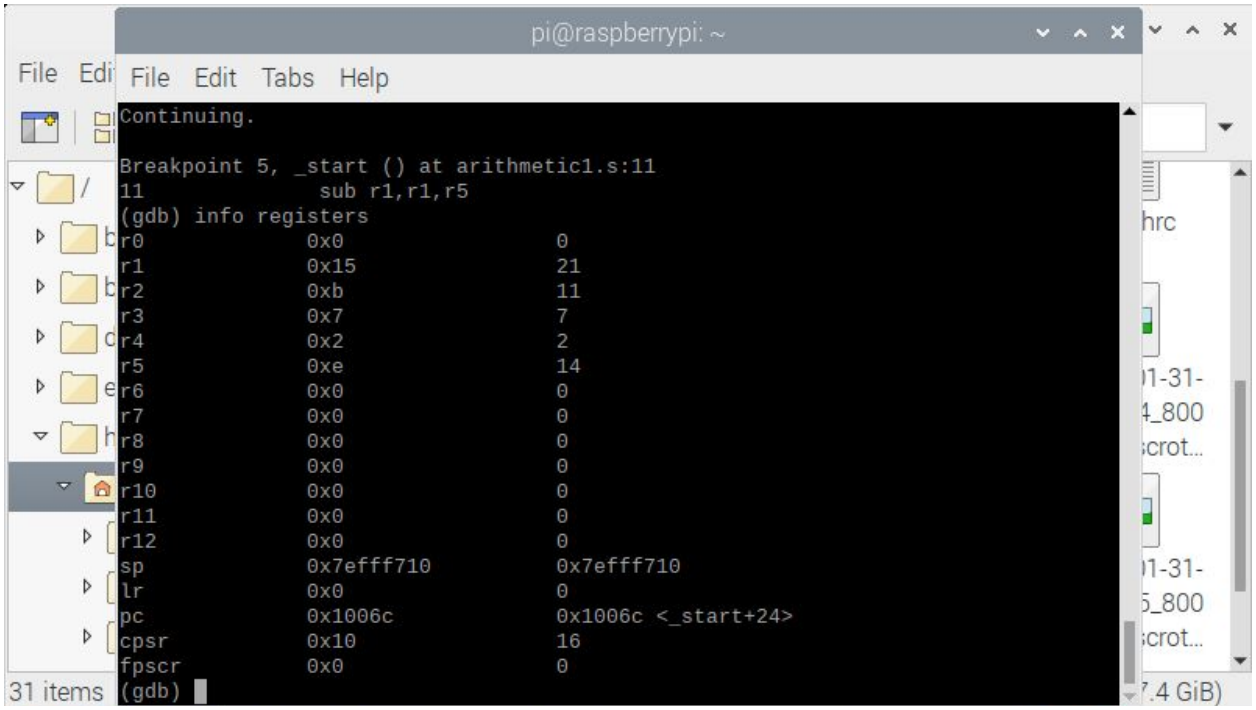
Next, I tackled arithmetic1. Below is a screenshot of the source code before I compiled it..



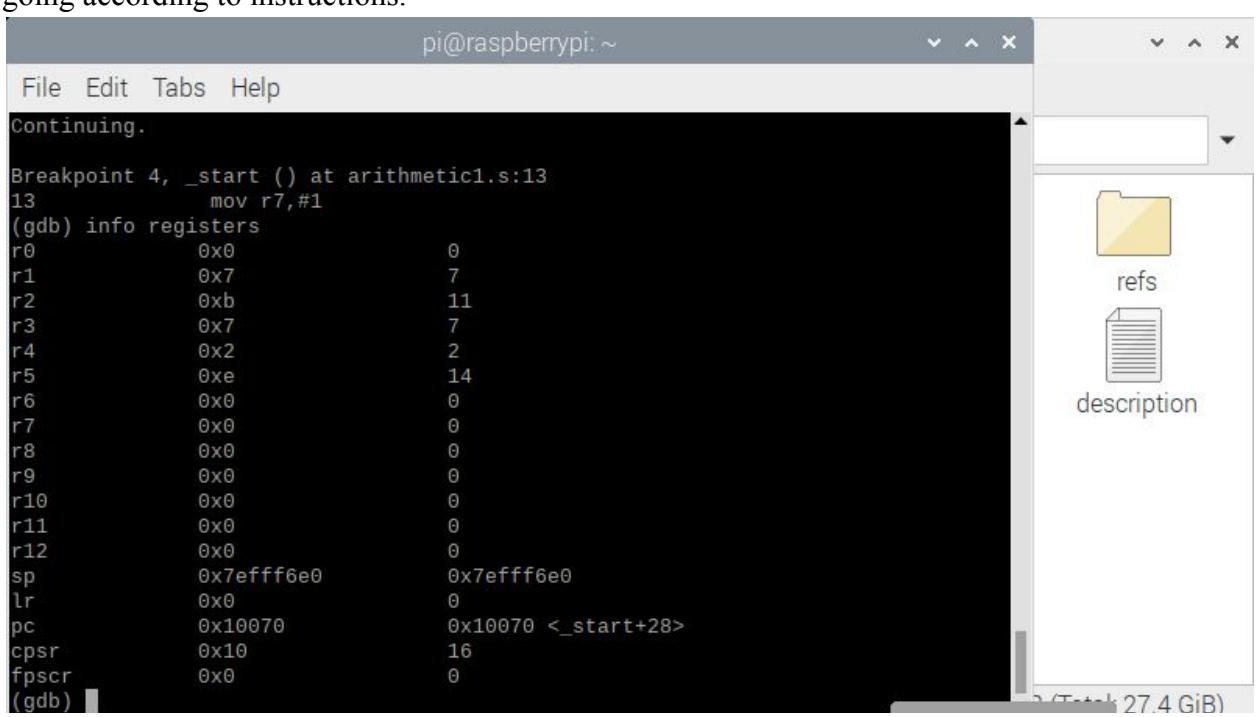
I loaded r1 with A, r2 with B, r3 with C, and finally r4 with D. Next I added r1 and r2, storing the outcome in r1. After, I multiplied r3 by r4, storing the value in r5. Lastly, I subtracted r5 from r1, storing the outcome in r1. Next, after loading the program, I checked the debugger and added in breakpoints to confirm that this is what was actually occuring in the registers.



Above we can see each registers r1 through r4 starting to be loaded with the values for A to D.



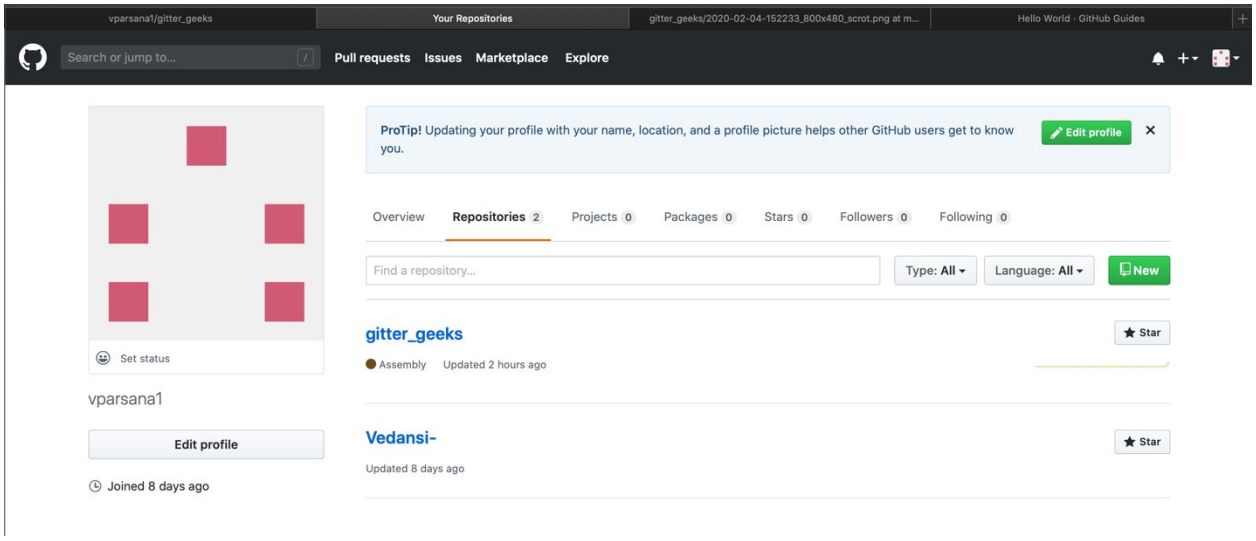
Here is how the registers look after executing the add and multiply instructions. As we can see, 10 plus 11 is 21, and 21 is in r1. 7 times 2 is 14, and 14 is stored in r5, so thus far everything is going according to instructions.



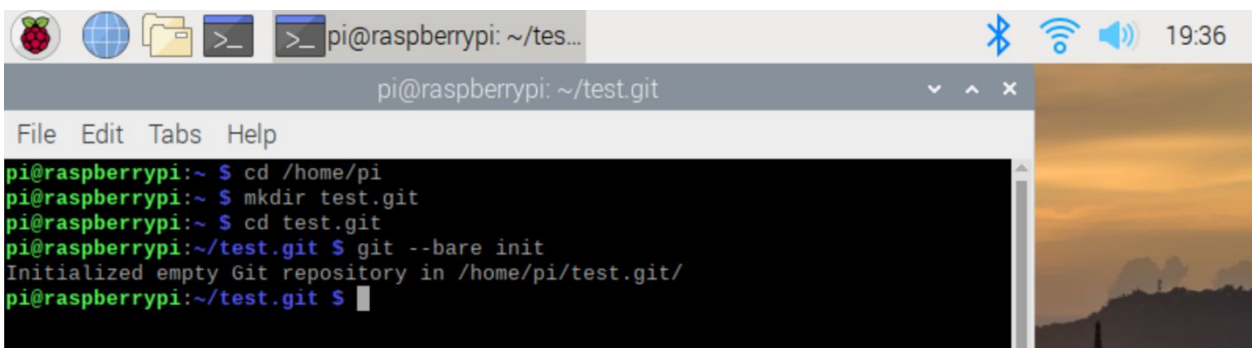
Lastly, we can see that 21 minus 14 equals 7, and 7 is stored in r1. Finally, cross checking with a calculator, we get (10 + 11) - (7 * 2) = 7.

Vedansi Parsana

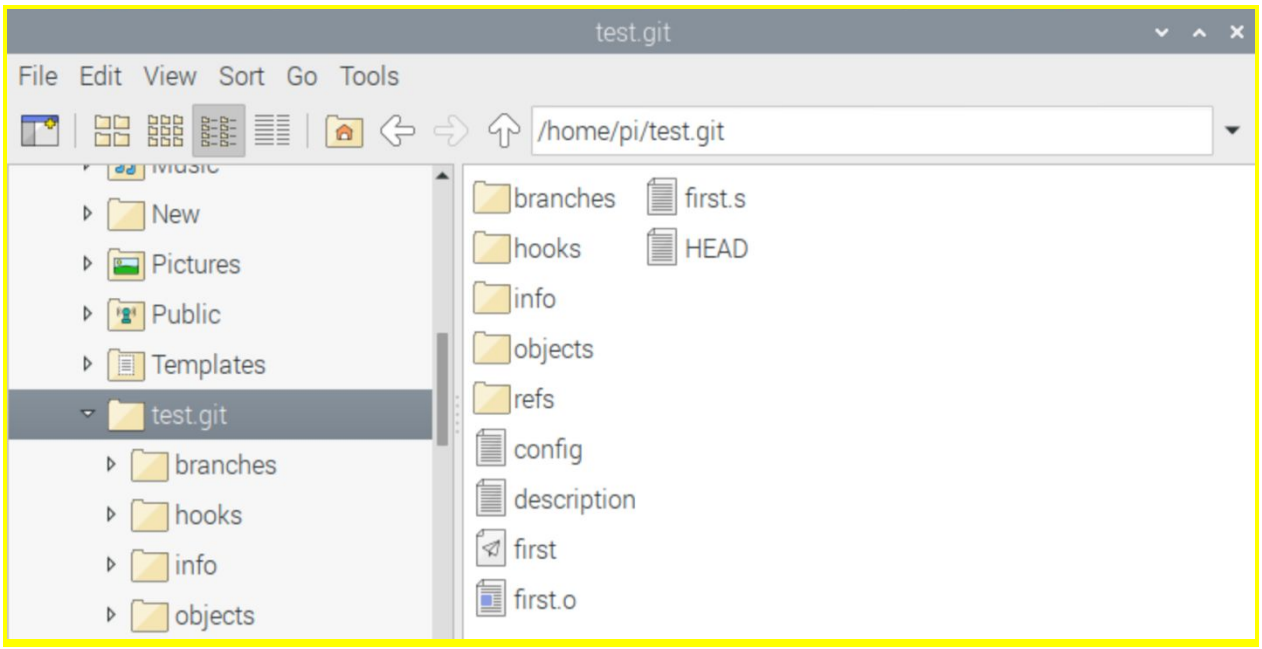
PI: First we made a repository in github, gitter_geeks is the name of our team and that is the repository which we use.



Using the steps given to us we generated a key and pasted it in our github account under ssh key. We also made a directory named test.git to save the following files.

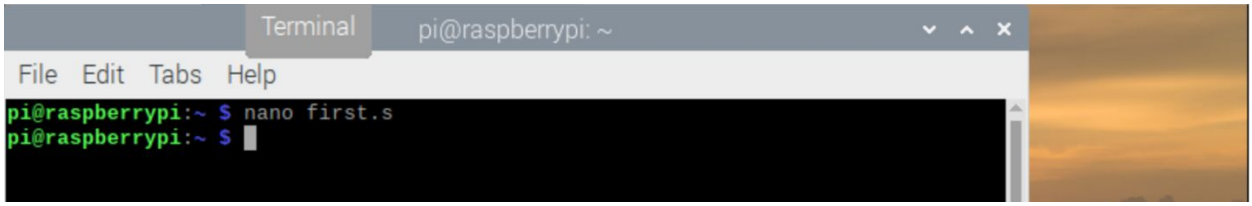


Here is how it looks when we check the repository, we see that three new files were created. In this case, it looks like:

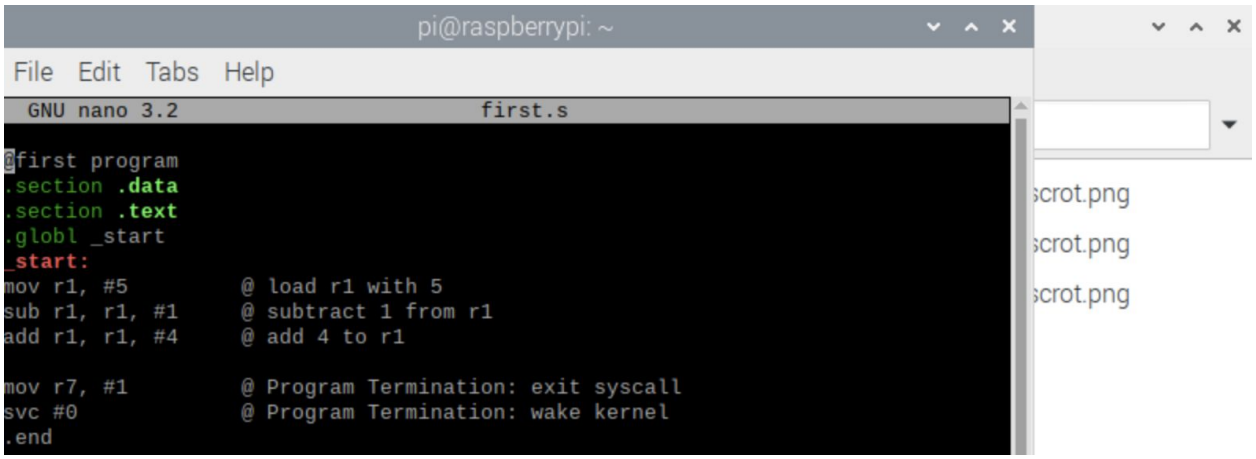


ARM Assembly Programming

Part 1 – After making the repository in GitHub for our code and connecting our Raspberry Pi to the repository at the GitHub using Git we started the following program.



With the help of the nano text editor by way of the terminal window, we saved the following code in the file.



```
pi@raspberrypi: ~
File  Edit  Tabs  Help
pi@raspberrypi:~ $ as -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ ./first
pi@raspberrypi:~ $ as -g -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ gdb first
GNU gdb (Raspbian 8.2.1-2) 8.2.1
Copyright (C) 2018 Free Software Foundation, Inc.
```

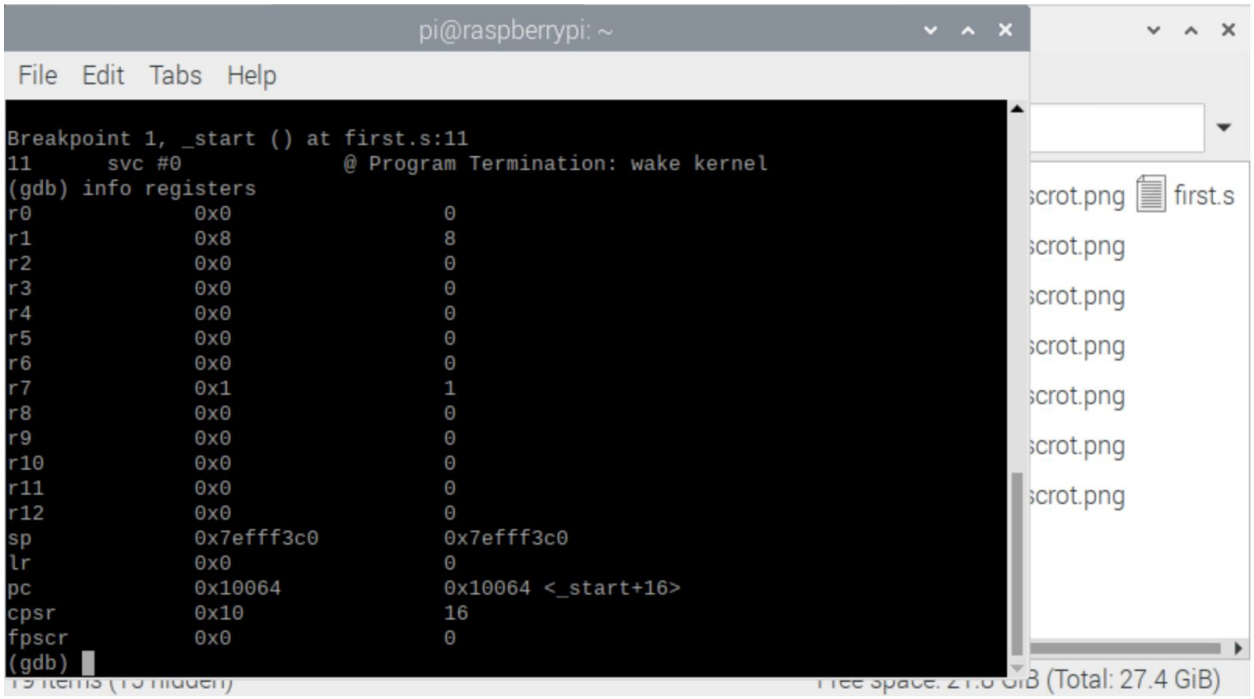
After creating the file we assembled the file and then linked the file to get an executable. After executing the file we ran the program. The result was truly expected, we did not see anything in the output because everything in the program takes place in the registry.

After that we begin utilizing the GDB to see the how the program is being executed. We used -g flag to reassemble the program to run GDB.

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from first...done.
(gdb) list
1       @first program
2       .section .data
3       .section .text
4       .globl _start
5       _start:
6       mov r1, #5          @ load r1 with 5
7       sub r1, r1, #1      @ subtract 1 from r1
8       add r1, r1, #4      @ add 4 to r1
9
10      mov r7, #1          @ Program Termination: exit syscall
(gdb)
```

```
8       add r1, r1, #4      @ add 4 to r1
9
10      mov r7, #1          @ Program Termination: exit syscall
(gdb) b 11
Breakpoint 1 at 0x10064: file first.s, line 11.
(gdb) run
Starting program: /home/pi/first
```

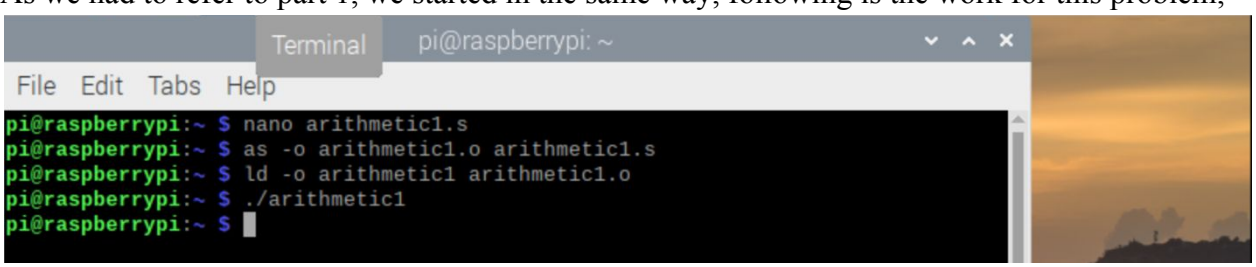
We then set breakpoint at line 11 to stop the program preceding the execution of the respective line. Then we gave GDB the command b 11, and then we ran the program. After that we gave the command run to look at the breakpoint.



We used the command "info registers" to see the generated values in the registers before executing line 11, Register 1 has a value of 8 because the register was first set to 5, then it was 4 because 1 was subtracted from it, and then 4 was added to that value, resulting it to 8.

Part 2 - Write a program that calculates the following expression: A = (A + B) − (C * D), where A=10, B=11, C=7, and D=2

As we had to refer to part 1, we started in the same way, following is the work for this problem;



As in Part 1, With the help of the nano text editor by way of the terminal window, we saved the following code in the file and the program looks as shown below. After creating the file, we assembled the file and then linked the file to get an executable. After executing the file, we ran the program. The result was truly expected, we did not see anything in the output because the entire program takes place in the registry. Therefore, we will not receive any output. The same as we did earlier.

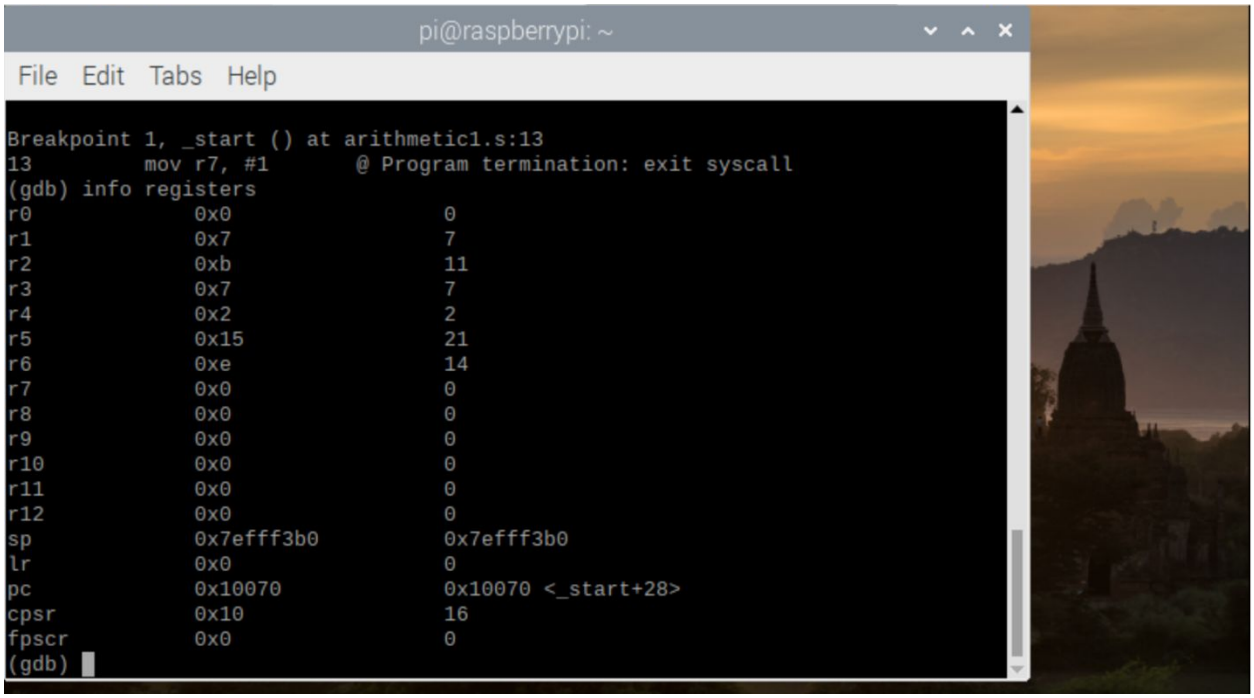
```
pi@raspberrypi: ~
File  Edit  Tabs  Help
  GNU nano 3.2                    arithmetic1.s

.section .data
.section .text
.global _start
_start:
   mov r1, #10
   mov r2, #11
   mov r3, #7
   mov r4, #2
   add r5, r1, r2   @ X = A+B
   mul r6, r3, r4   @ Y = C*D
   sub r1, r5, r6   @ A = X-Y

   mov r7, #1       @ Program termination: exit syscall
   svc #0           @ Program termination: wake kernel
.end
```

After that we begin utilizing the GDB to see how our First Program was being executed. We used -g flag to reassemble the program to run GDB, as we did before.

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from arithmetic1...done.
(gdb) run
Starting program: /home/pi/arithmetic1
[Inferior 1 (process 10695) exited normally]
(gdb)
```

```
pi@raspberrypi: ~
File  Edit  Tabs  Help
(gdb) b 13
Note: breakpoints 1 and 2 also set at pc 0x10070.
Breakpoint 4 at 0x10070: file arithmetic1.s, line 13.
(gdb) run
Starting program: /home/pi/arithmetic1

Breakpoint 1, _start () at arithmetic1.s:13
13          mov r7, #1       @ Program termination: exit syscall
(gdb) info registers
r0             0x0                 0
r1             0x7                 7
r2             0xb                 11
r3             0x7                 7
r4             0x2                 2
r5             0x15                21
r6             0xe                 14
r7             0x0                 0
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
r12            0x0                 0
sp             0x7efff3b0          0x7efff3b0
```

Here we set the breakpoint to line 14 , so to debug it through every step before we end the code.



So the above registers states that - Registers 2 is 11, 3 is 7, 4 is 2 which is completely expected because we had the assigned values. Register 5 stores the value of 22 - the result of our initial A added to B (10+11). Register 6 stores the value as 14 which is the result of C*D (7*2). Register 1, stores the final value of A which is 7 which is the expected result of the given equation.

Michael Knight

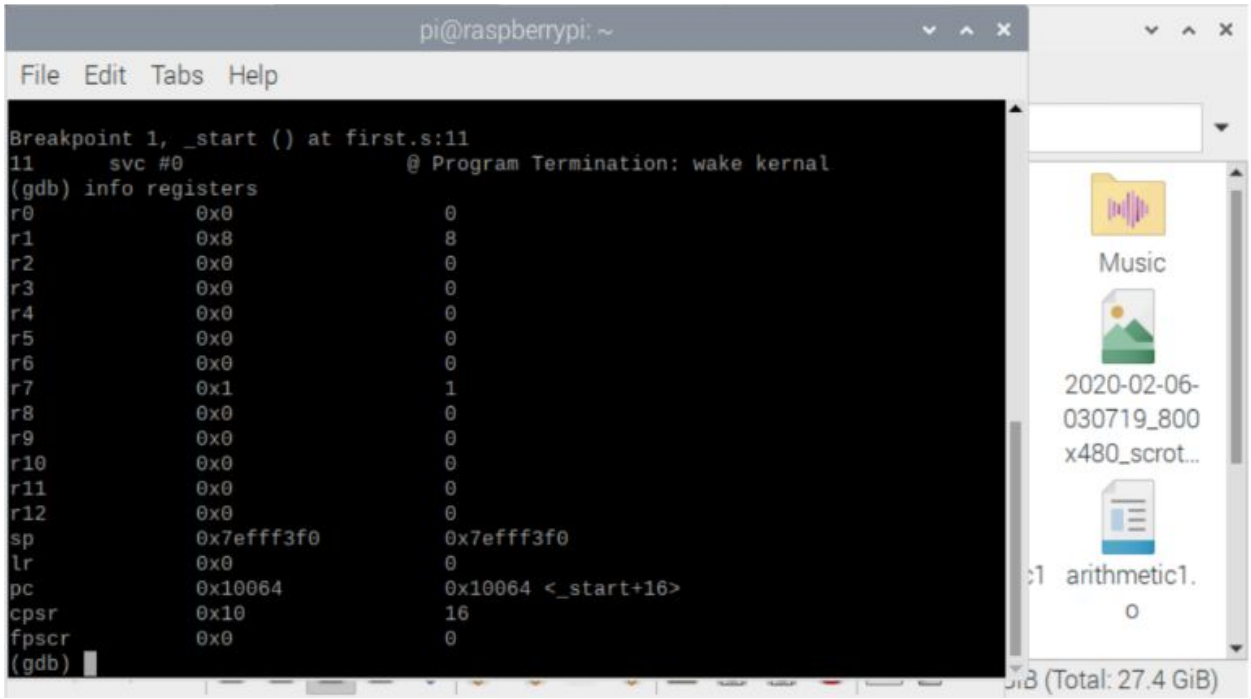
Project A1 Wrote the program and as instructed.

```
pi@raspberrypi: ~
File  Edit  Tabs  Help
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from first...done.
(gdb) list
1       @first program
2       .section .data
3       .section .text
4       .globl _start
5       _start:
6       mov r1, #5              @ load r1 with 5
7       sub r1, r1, #1          @ subtract 1 from r1
8       add r1,r1, #4           @ add 4 to r1
9
10      mov r7, #1              @ Progeam Termination: exit syscall
(gdb) b11
Undefined command: "b11".  Try "help".
(gdb) run
Starting program: /home/pi/first
[Inferior 1 (process 1261) exited normally]
(gdb) info registers
The program has no registers now.
(gdb)
```

After writing it and running it, no output was shown because we've only assigned values to registers and we haven't run any instructions to produce an output.

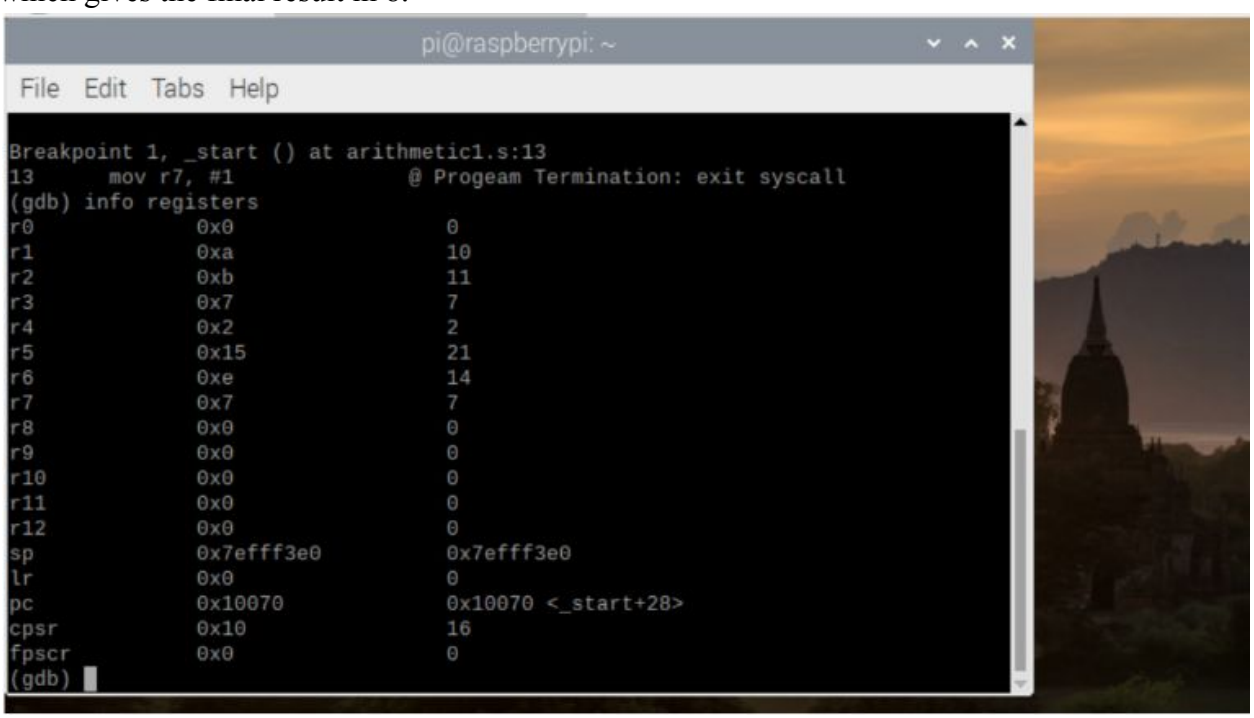
```
pi@raspberrypi: ~
File  Edit  Tabs  Help
10      mov r7, #1              @ Progeam Termination: exit syscall
(gdb) b11
Undefined command: "b11".  Try "help".
(gdb) run
Starting program: /home/pi/first
[Inferior 1 (process 1261) exited normally]
(gdb) info registers
The program has no registers now.
(gdb) quit
pi@raspberrypi:~ $ nano
Use "fg" to return to nano.

[1]+  Stopped                 nano
pi@raspberrypi:~ $ nano
pi@raspberrypi:~ $ nano
pi@raspberrypi:~ $ as -o arithmetic1.o arithmetic1.s
pi@raspberrypi:~ $ ld -o arithmetic1 arithmetic.o
ld: cannot find arithmetic.o: No such file or directory
pi@raspberrypi:~ $ ld -o arithmetic1 arithmetic1.o
pi@raspberrypi:~ $ ./arithmetic1
pi@raspberrypi:~ $ as -g -o arithmetic1.o arithmetic1.s
pi@raspberrypi:~ $ ld -o arithmetic1 arithmetic1.o
pi@raspberrypi:~ $
```

Picture is just assembling and taking that to the linker.



Above picture is one of running "info registers"

Here we see the value placed on r1 as 8 which is what is to be expected since you first assigned the value of 5 into r1 then subtracted 1 away from it which results in 4, but then we added 4 to it which gives the final result in 8.

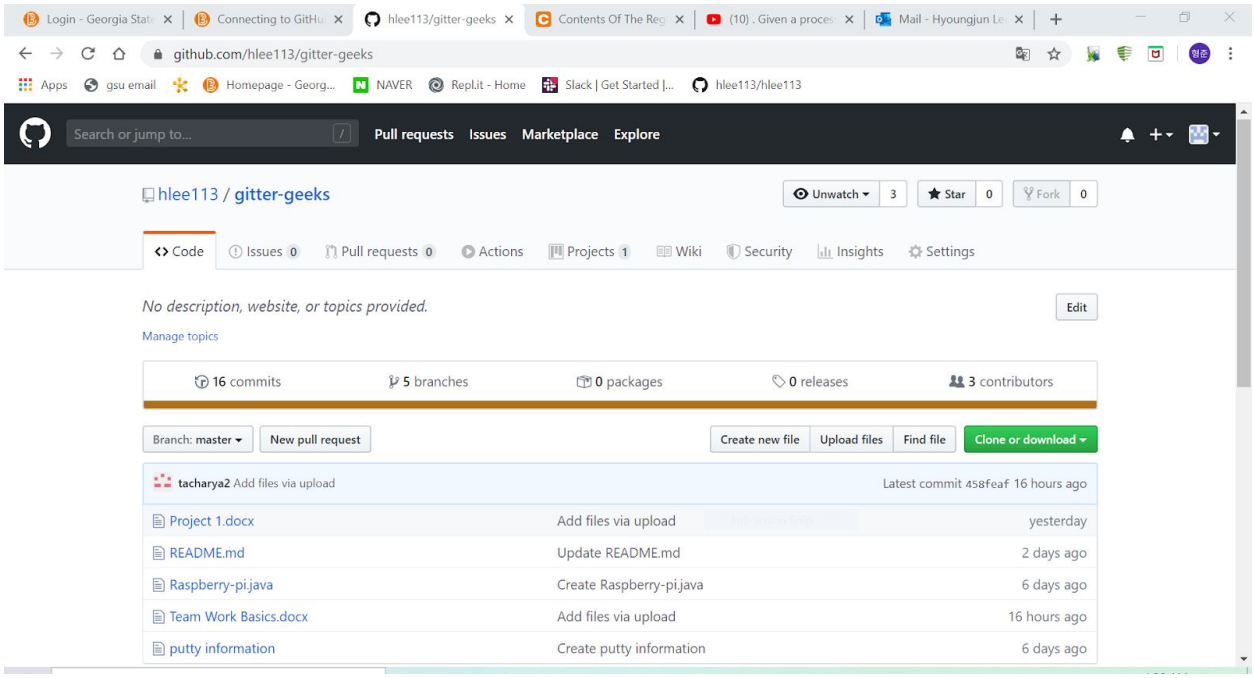


Above picture is one of the program arithmetic1 that we were instructed to write while using first.s as a reference. We see r1 stored with the value of 10, r2 stored with 11, r3 stored with 7 r4 stored with 2. R5 is stored with the value of r1 and r2 added together. R6 is stored with the value of r3 and r4 multiplied together. R7 is stored with the value of r5 subtracted by r6. So our answer

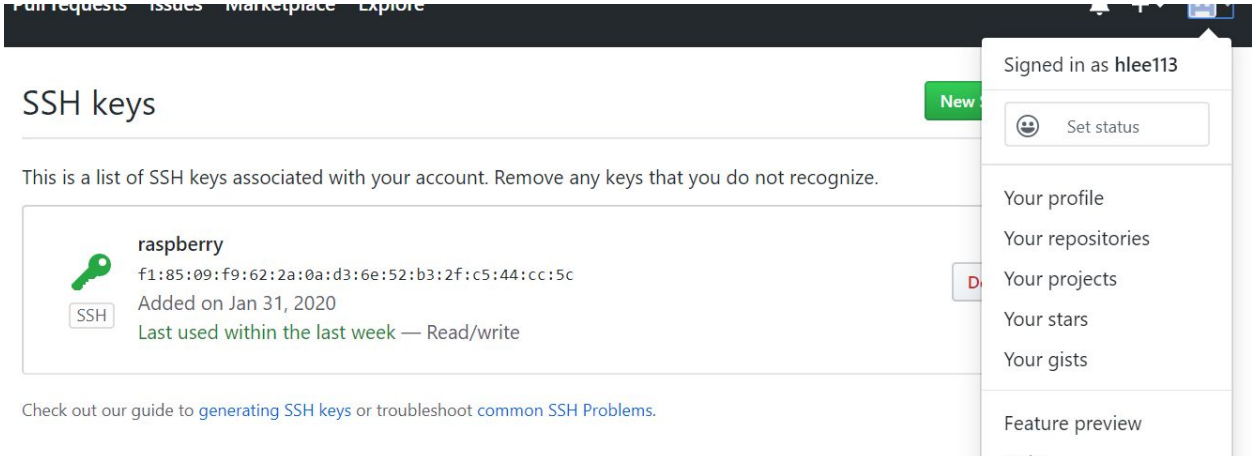
is the value that is held in r7. When this program is run we still see no results because we're only storing values in registers and not executing any output.

Hyoungjun Lee

We named gitter-geeks as our group name, then we made github to organize our stuff



And generate SSH and connect to individual account



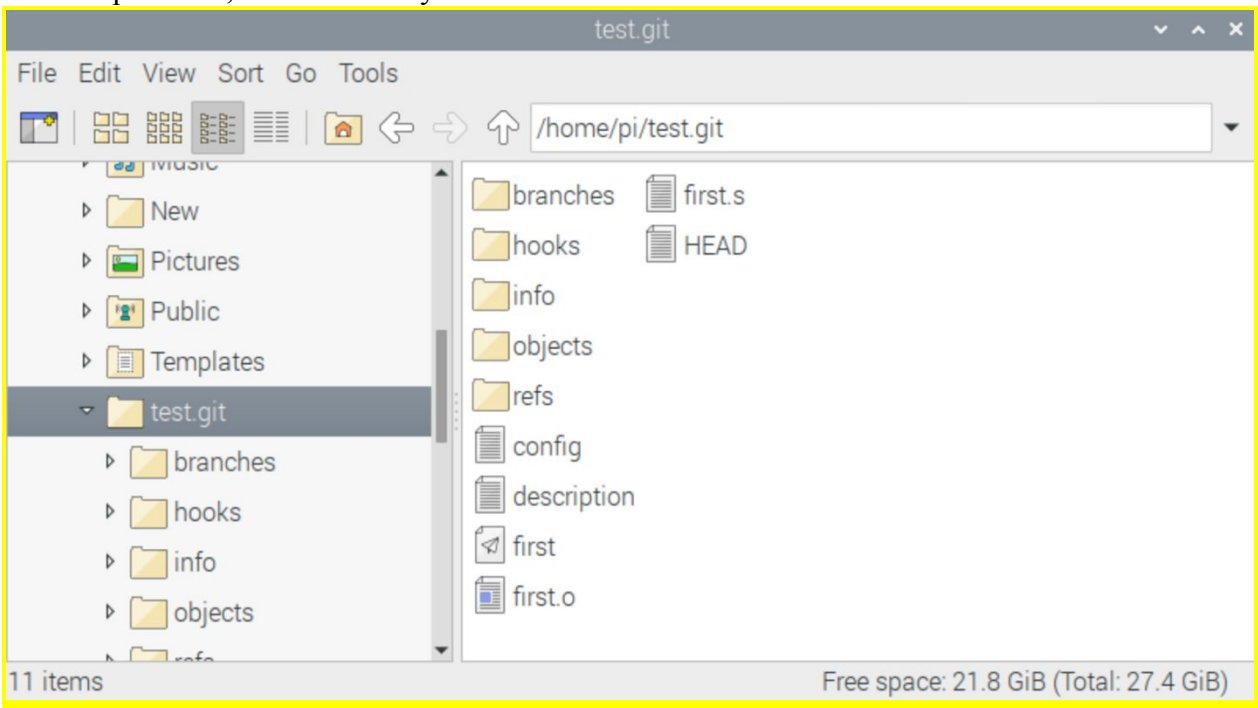
After generate SSH, First I connected to github with SSH, successfully connected as shown below

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jan 31 09:17:11 2020
pi@raspberrypi1:~ $ ssh -T git@github.com
Warning: Permanently added the RSA host key for IP address '192.30.253.112' to t
he list of known hosts.
git@github.com: Permission denied (publickey).
pi@raspberrypi1:~ $ sudo -i
root@raspberrypi1:~# ssh -T git@github.com
Enter passphrase for key '/root/.ssh/id_rsa':
git@github.com: Permission denied (publickey).
root@raspberrypi1:~# ssh -T git@github.com
Warning: Permanently added the RSA host key for IP address '140.82.113.4' to the
 list of known hosts.
Enter passphrase for key '/root/.ssh/id_rsa':
Hi hlee113! You've successfully authenticated, but GitHub does not provide shell
 access.
root@raspberrypi1:~#
```

To test, make test.git file into home/pi

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Feb  5 08:01:28 2020 from 192.168.1.89
pi@raspberrypi1:~ $ sudo -i
root@raspberrypi1:~# cd/home/pi
-bash: cd/home/pi: No such file or directory
root@raspberrypi1:~# cd /home/pi
root@raspberrypi1:/home/pi# mkdir test.git
mkdir: cannot create directory 'test.git': File exists
root@raspberrypi1:/home/pi# cd test.git
root@raspberrypi1:/home/pi/test.git# git --bare init
Reinitialized existing Git repository in /home/pi/test.git/
root@raspberrypi1:/home/pi/test.git#
```

After steps above, this is what my folder looks like



ARM Assembly Programming

Part 1 – After making the repository in GitHub for our code and connecting our Raspberry Pi to the repository at the GitHub using Git we started the following program.

We use nano text editor to save a file inside and edit.

```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
Reading state information... Done
git is already the newest version (1:2.20.1-2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@raspberrypi1:~# ssh -T git@github.com
Enter passphrase for key '/root/.ssh/id_rsa':
git@github.com: Permission denied (publickey).
root@raspberrypi1:~# ssh -T git@github.com
Enter passphrase for key '/root/.ssh/id_rsa':
Hi hlee113! You've successfully authenticated, but GitHub does not provide shell
 access.
root@raspberrypi1:~# nano first.s
root@raspberrypi1:~# as -g -o first.o first.s
first.s: Assembler messages:
first.s:5: Error: ARM register expected -- `mov rl,#5'
first.s:6: Error: ARM register expected -- `sub rl,rl,#1'
first.s:7: Error: ARM register expected -- `add rl,rl,#4'
root@raspberrypi1:~# git remote remove origin
fatal: No such remote: 'origin'
root@raspberrypi1:~# git remote add origin git@github.com: hlee113/gitter-geeks.
```

```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
  GNU nano 3.2                    first.s                    Modified

x@first program
.section .data
.section .text
.globl _start
_start:
        mov rl, #5      @load rl with 5
        sub rl, rl, #1  @subtract 1 from rl
        add rl, rl, #4  @add 4 to rl

        mov r7, #1      @program Termination: exit syscall
        svc #0          @Program Termination: wake kernel
      .end
```

We create file and assembled file , also linked to executable , even after we run the code, we can not see anything because all the programs takes place into registry

```
pi@raspberrypi: ~
File  Edit  Tabs  Help
pi@raspberrypi:~ $ as -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ ./first
pi@raspberrypi:~ $ as -g -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ gdb first
GNU gdb (Raspbian 8.2.1-2) 8.2.1
Copyright (C) 2018 Free Software Foundation, Inc.
```

We used the gdb command (debugging) to see how the program is being executed. We used -g flag to reassemble the program to run GDB.

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from first...done.
(gdb) list
1       @first program
2       .section .data
3       .section .text
4       .globl _start
5       _start:
6       mov r1, #5          @ load r1 with 5
7       sub r1, r1, #1      @ subtract 1 from r1
8       add r1, r1, #4      @ add 4 to r1
9
10      mov r7, #1          @ Program Termination: exit syscall
(gdb)
```

This is just for github pulled

```
pi@raspberrypi1: ~
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
root@raspberrypi1:/home/pi/test.git# git checkout master
Already on 'master'
root@raspberrypi1:/home/pi/test.git# git pull
Enter passphrase for key '/root/.ssh/id_rsa':
warning: no common commits
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 38 (delta 5), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (38/38), done.
From github.com:hlee113/gitter-geeks
 * [new branch]      Michael-Knight -> origin/Michael-Knight
 * [new branch]      Tek-Sarma      -> origin/Tek-Sarma
 * [new branch]      Vedansi        -> origin/Vedansi
 * [new branch]      Zoe-a-Kosmicki -> origin/Zoe-a-Kosmicki
 * [new branch]      master         -> origin/master
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>
```

We set the breakpoint at 11, When we run the program, the program execution will stop right before line 11 is executed

```
pi@raspberrypi: ~
File  Edit  Tabs  Help

Breakpoint 1, _start () at first.s:11
11      svc #0              @ Program Termination: wake kernel
(gdb) info registers
r0             0x0                 0
r1             0x8                 8
r2             0x0                 0
r3             0x0                 0
r4             0x0                 0
r5             0x0                 0
r6             0x0                 0
r7             0x1                 1
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
r12            0x0                 0
sp             0x7efff3c0          0x7efff3c0
lr             0x0                 0
pc             0x10064             0x10064 <_start+16>
cpsr           0x10                16
fpscr          0x0                 0
(gdb)
```

We used command info register to verify literal value in register, Register 1 has a value of 8 , because at the first register set to value of 5, and then it was 4 because 1 was subtracted from it, then 4 was added to that value, which made it result to 8.

Part 2 - Write a program that calculates the following expression: A = (A + B) − (C * D), where A=10, B=11, C=7, and D=2

Start with nano editor as we did in part 1

We saved following code in arithmetics1.s and then assembled file then linked to get executable, we ran the program, as we expected we could not see any result

```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
pi@raspberrypi1:~ $ nano arithmetic.s
pi@raspberrypi1:~ $ as -o arithmetic1.o arithmetic1.s
Assembler messages:
Error: can't open arithmetic1.s for reading: No such file or directory
pi@raspberrypi1:~ $ nano arithmetic1.s
pi@raspberrypi1:~ $ as -o arithmetic1.o arithmetic1.s
pi@raspberrypi1:~ $ ld -o arithmetic1 arithmetic1.o
pi@raspberrypi1:~ $ ./arithmetic1
pi@raspberrypi1:~ $
```

```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
  GNU nano 3.2                    arithmetic1.s

.section .data
.section .text
.global _start
_start:
  mov r1, #10
  mov r2, #11
  mov r3, #7
  mov r4, #2
  add r5, r1, r2  @ X = A + B
  mul r6, r3, r4  @ Y = C*D
  sub r1, r5, r6  @ A = X-Y

  mov r7, #1      @ Program termination: exit syscall
  svc #0          @ Program termination: wake kernel
```

We used gdb (debugging) command again to see how this program being executed we used -g flag to reassemble and run GDB

```
pi@raspberrypi1: ~   pi@raspberrypi1: ~
File  Edit  Tabs  Help
pi@raspberrypi1:~ $ ld -o arithmetic1 arithmetic1.o
pi@raspberrypi1:~ $ gdb arithmetic1
GNU gdb (Raspbian 8.2.1-2) 8.2.1
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
```

```
pi@raspberrypi1: ~   pi@raspberrypi1: ~
File  Edit  Tabs  Help

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from arithmetic1...done.
(gdb) run
Starting program: /home/pi/arithmetic1
[Inferior 1 (process 4181) exited normally]
(gdb) list
1       .section .data
2       .section .text
3       .global _start
4       _start:
5         mov r1, #10
6         mov r2, #11
7         mov r3, #7
8         mov r4, #2
9         add r5, r1, r2  @ X = A + B
10        mul r6, r3, r4  @ Y = C*D
(gdb)
```

We set the breakpoint at line 14, debug will go through every step that we set the point

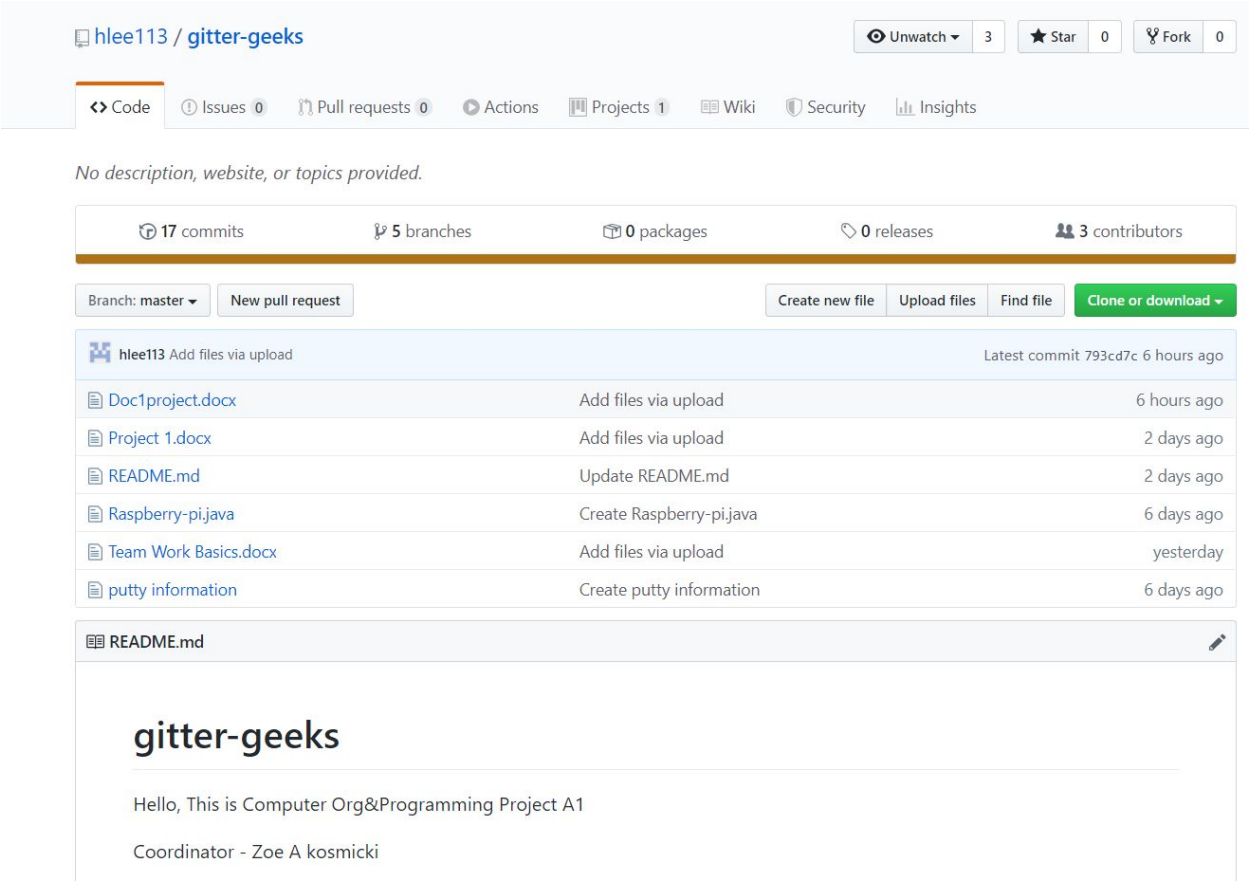
```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/arithmetic1

Breakpoint 1, _start () at arithmetic1.s:13
13          mov r7, #1      @ Program termination: exit syscall
(gdb) info registers
r0             0x0                 0
r1             0x7                 7
r2             0xb                 11
r3             0x7                 7
r4             0x2                 2
r5             0x15                21
r6             0xe                 14
r7             0x0                 0
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
```

```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
r0             0x0                 0
r1             0x7                 7
r2             0xb                 11
r3             0x7                 7
r4             0x2                 2
r5             0x15                21
r6             0xe                 14
r7             0x0                 0
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
r12            0x0                 0
sp             0x7efff3b0          0x7efff3b0
lr             0x0                 0
pc             0x10070             0x10070 <_start+28>
cpsr           0x10                16
fpscr          0x0                 0
(gdb)
```

These are the registers that shows r2 (register 2) is 11, 3 for 7 , 4 for 2
This is totally what we expected because A = (A + B) − (C *D),
Where A = 10 B =11 C = 7 D =2
Register 6 stores result of value (C * D) = 14 , register 1 stores final value of 7.

Tek Acharya

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



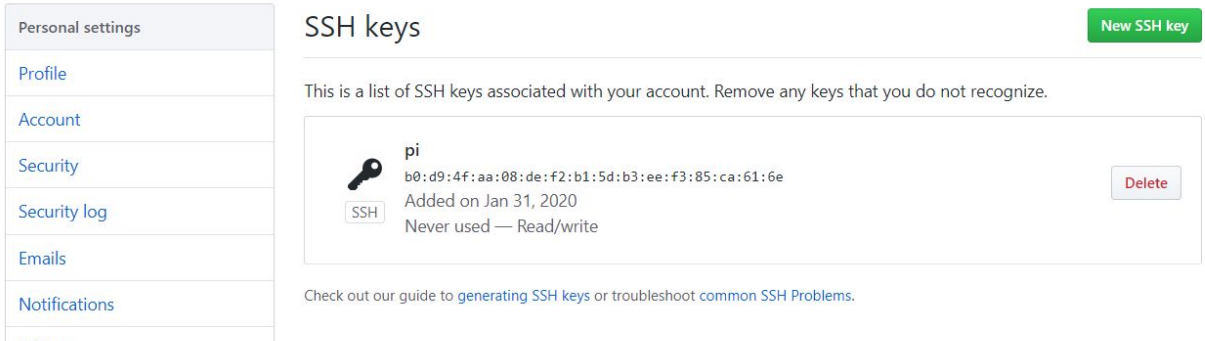
We, then connected our raspberry pi with the GitHub using SSH connection

```
pi@raspberrypi: ~
File  Edit  Tabs  Help
pi@raspberrypi:~ $ sudo -i
root@raspberrypi:~# sudo apt-get install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.20.1-2+deb10u1).
The following packages were automatically installed and are no longer required:
  alsa-base gstreamer0.10-alsa gstreamer0.10-plugins-base
  libgstreamer-plugins-base0.10-0 libgstreamer0.10-0 libva-wayland2
  libxfce4util-bin libxfce4util-common libxfce4util7 libxfconf-0-2 pimixer
  point-rpi xfconf
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@raspberrypi:~# git config --global user.name Tek Acharya
root@raspberrypi:~# git config --global user.email akash_sharma430@yahoo.com
root@raspberrypi:~# sudo apt-get install ssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
ssh is already the newest version (1:7.9p1-10+deb10u1).
The following packages were automatically installed and are no longer required:
  alsa-base gstreamer0.10-alsa gstreamer0.10-plugins-base
  libgstreamer-plugins-base0.10-0 libgstreamer0.10-0 libva-wayland2
```

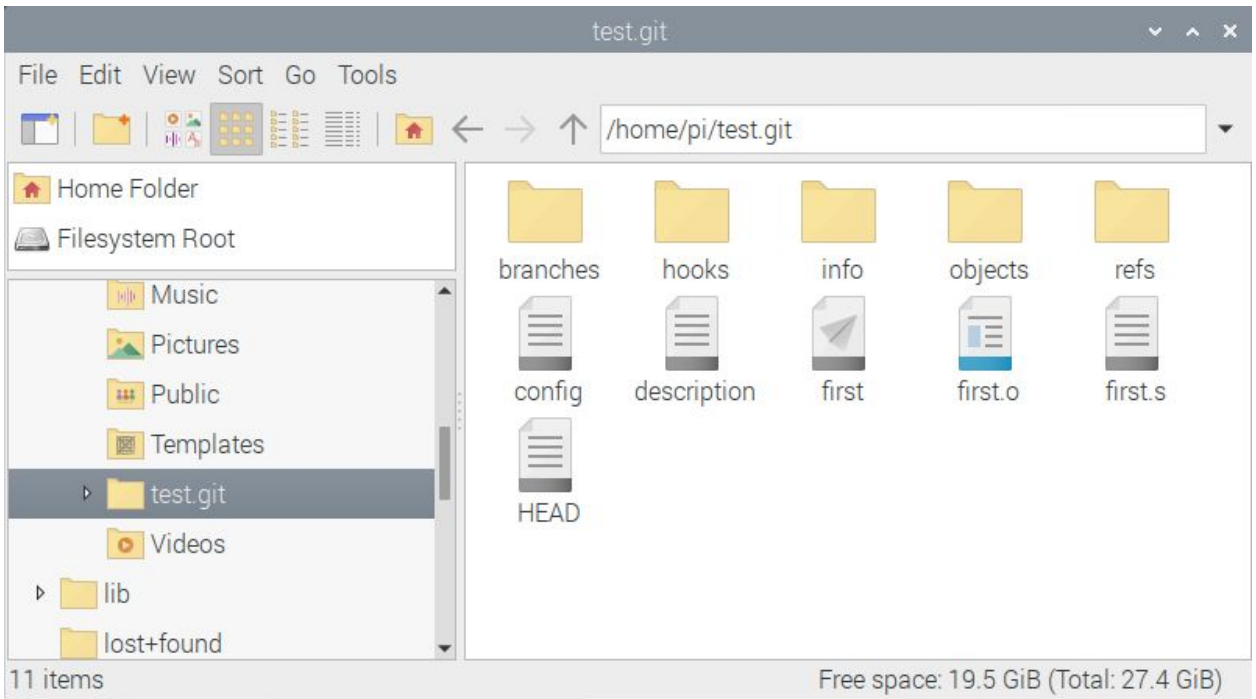
```
a:61:6e

SSH Problems.
```

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

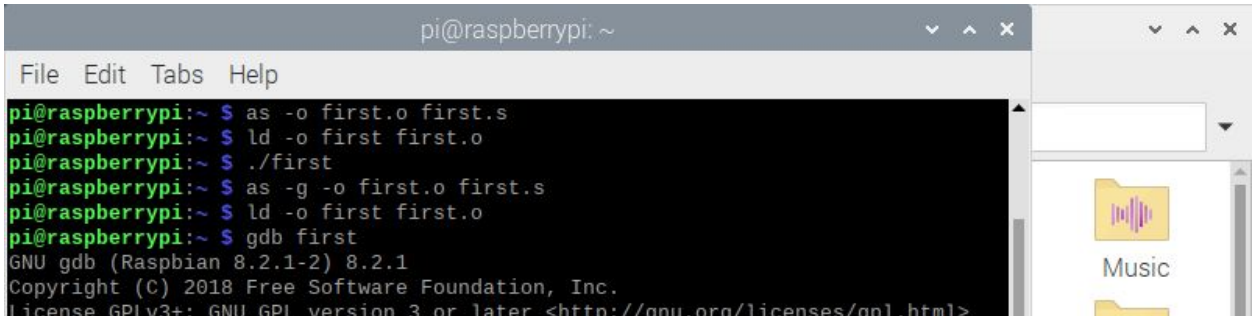
```
pi@raspberrypi1: ~
Linux raspberrypi1 4.19.50-v7+ #896 SMP Thu Jun 20 16:11:44 BST 2019 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jan 31 09:17:11 2020
pi@raspberrypi1:~ $ ssh -T git@github.com
Warning: Permanently added the RSA host key for IP address '192.30.253.112' to t
he list of known hosts.
git@github.com: Permission denied (publickey).
pi@raspberrypi1:~ $ sudo -i
root@raspberrypi1:~# ssh -T git@github.com
Enter passphrase for key '/root/.ssh/id_rsa':
git@github.com: Permission denied (publickey).
root@raspberrypi1:~# ssh -T git@github.com
Warning: Permanently added the RSA host key for IP address '140.82.113.4' to the
 list of known hosts.
Enter passphrase for key '/root/.ssh/id_rsa':
Hi hlee113! You've successfully authenticated, but GitHub does not provide shell
 access.
root@raspberrypi1:~#
```



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



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



We compile, assemble and link first.s assembly program. 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.

```
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from first...done.
(gdb) list
1       @first program
2       .section .data
3       .section .text
4       .global _start
5       _start:
6             mov r1, #5     @load r1 with 5
7             sub r1, #1     @subtract 1 from r1
8             add r1, r1, #4 @ add 4 to r1
9
10            mov r7, #1     @Program termination: Exit system call
(gdb)
```

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 11 using b 11 command and run the code to see what happens into the registers and found the following output.

```
pi@raspberrypi1: ~
File  Edit  Tabs  Help
  GNU nano 3.2                    first.s                    Modified

x@first program
.section .data
.section .text
.globl _start
_start:
        mov rl, #5      @load rl with 5
        sub rl, rl, #1  @subtract 1 from rl
        add rl, rl, #4  @add 4 to rl

        mov r7, #1      @program Termination: exit syscall
        svc #0          @Program Termination: wake kernel
      .end

                        [ Read 12 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text^T To Spell  ^_ Go To Line
```

```
File  Edit  Tabs  Help

Breakpoint 1, _start () at first.s:11
11      svc #0              @ Program Termination: wake kernel
(gdb) info registers
r0             0x0                 0
r1             0x8                 8
r2             0x0                 0
r3             0x0                 0
r4             0x0                 0
r5             0x0                 0
r6             0x0                 0
r7             0x1                 1
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
r12            0x0                 0
sp             0x7efff3c0          0x7efff3c0
lr             0x0                 0
pc             0x10064             0x10064 <_start+16>
cpsr           0x10                16
fpscr          0x0                 0
(gdb)
```

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)

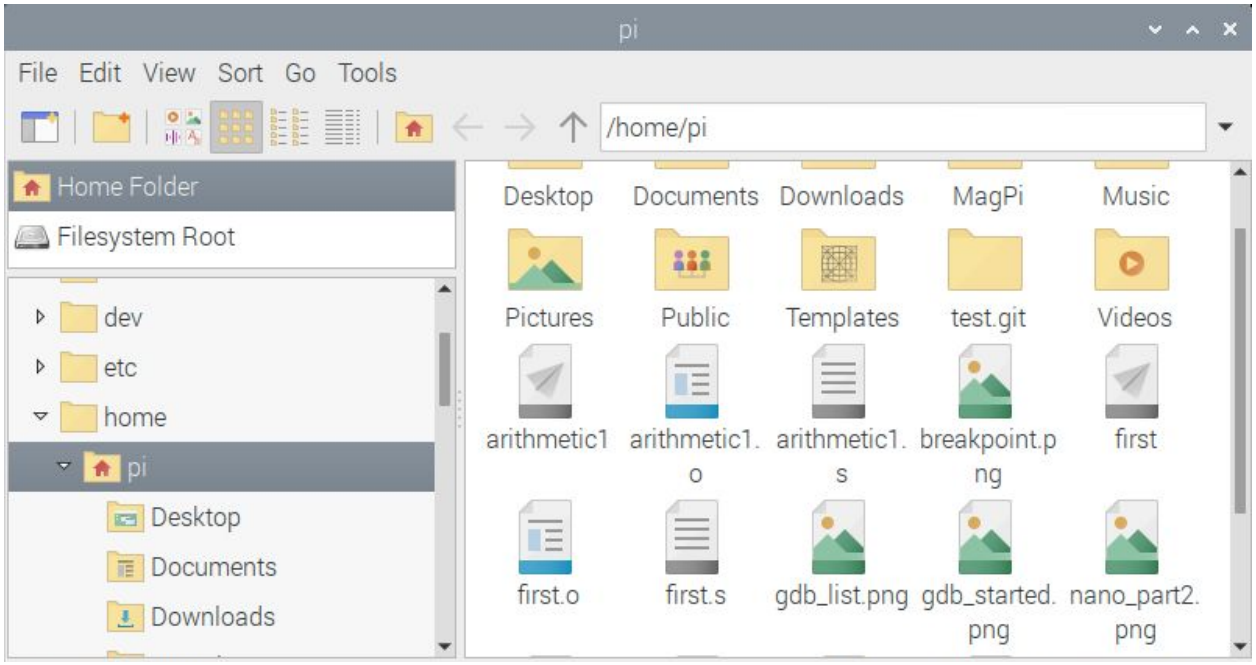
Here: A = 10, B = 11, C = 7, and D = 2.

We named our new program as arithmatic1.s

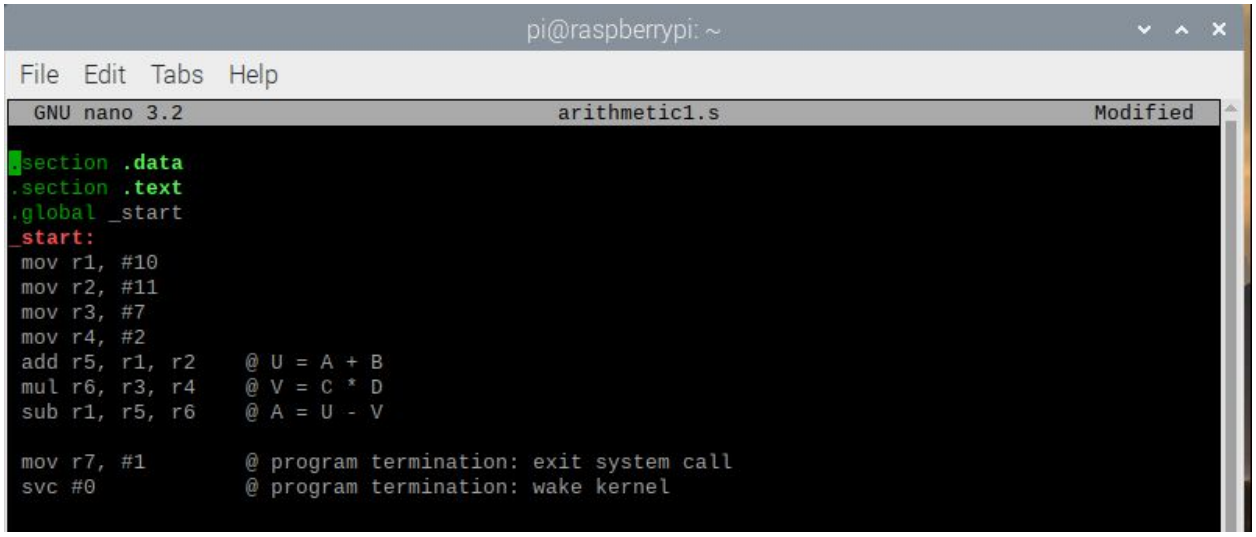
```
pi@raspberrypi: ~
File  Edit  Tabs  Help
pi@raspberrypi:~ $ as -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ ./first
pi@raspberrypi:~ $ as -g -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ gdb first
GNU gdb (Raspbian 8.2.1-2) 8.2.1
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
```

Music

Assembled, linked, and Debugging done



Executable created



Nano Arithmatic1.s compiled

```
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from arithmetic1...done.
(gdb) list
1       .section .data
2       .section .text
3       .global _start
4       _start:
5        mov r1, #10
6        mov r2, #11
7        mov r3, #7
8        mov r4, #2
9        add r5, r1, r2    @ U = A + B
10       mul r6, r3, r4    @ V = C * D
(gdb)
```

listed

```
pi@raspberrypi: ~
File  Edit  Tabs  Help

Breakpoint 1, _start () at arithmetic1.s:14
14      svc #0          @ program termination: wake kernel
(gdb) info registers
r0             0x0                 0
r1             0x7                 7
r2             0xb                 11
r3             0x7                 7
r4             0x2                 2
r5             0x15                21
r6             0xe                 14
r7             0x1                 1
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
r12            0x0                 0
sp             0x7efff3b0          0x7efff3b0
lr             0x0                 0
pc             0x10074             0x10074 <_start+32>
cpsr           0x10                16
fpscr          0x0                 0
(gdb)
```

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.

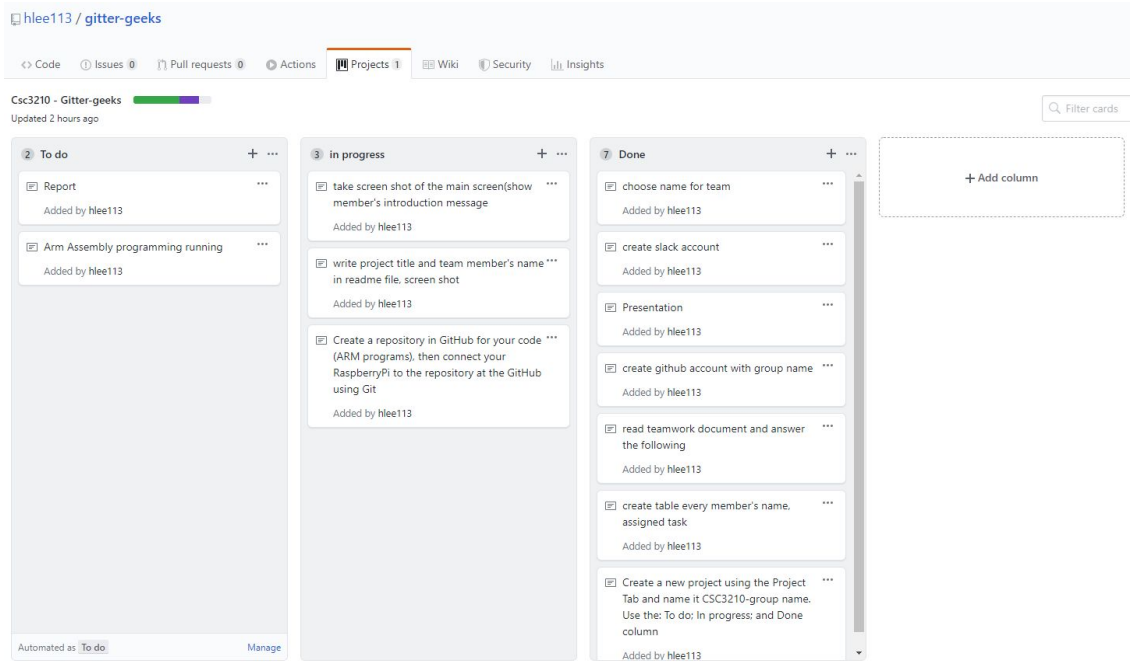
Appendix

LINKS

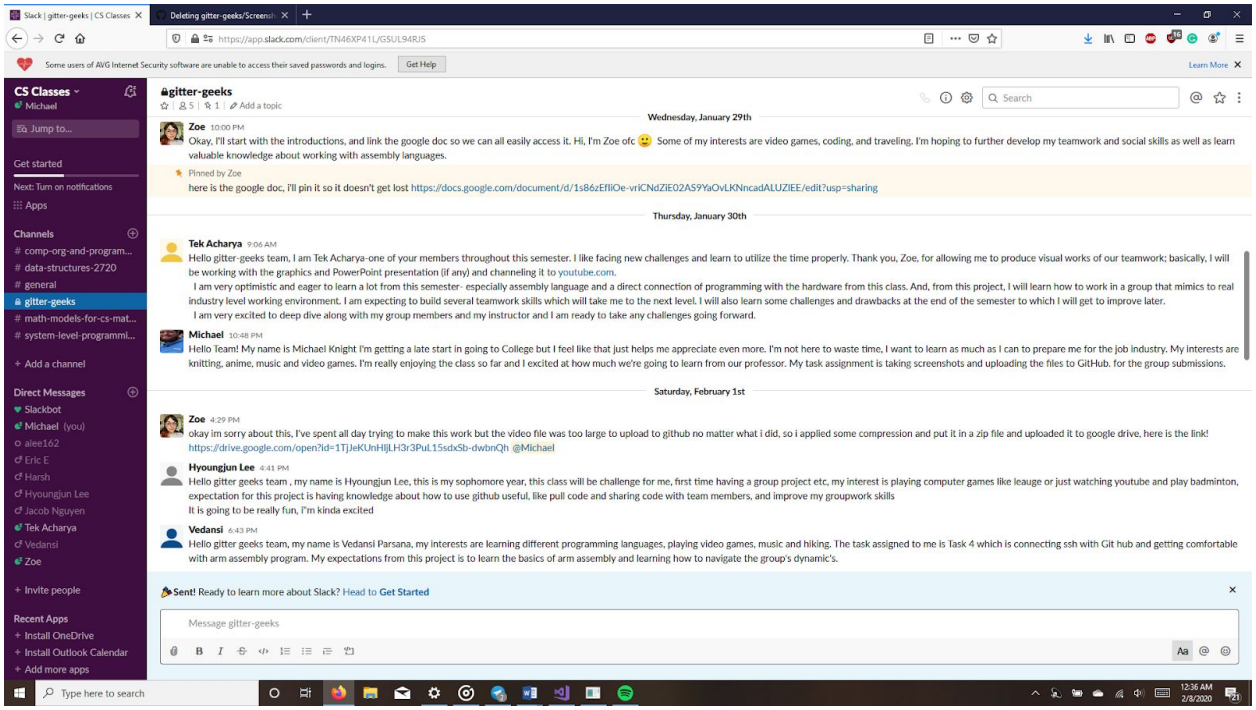
GitHub link: https://github.com/hlee113/gitter-geeks
YouTube link: https://www.youtube.com/watch?v=BED4Ckts9dM
Slack link: https://cs-classes-workspace.slack.com/archives/GSUL94RJ5

SCREENSHOTS

GitHub:



Slack Introductions:



Planning and Scheduling Timetable:

Names	Email	Task	Dependencies	Duration	Due Date	Notes
Hyoungjun Lee	hlee113@student.gsu.edu	Creating our GitHub account and verifying everyone has access to it and connected with their Raspberry Pis	Needs everyone to generate SSH keys and confirm they have access to the GitHub account	4 hours	2/1/20	100%
Michael Knight	mknight23@student.gsu.edu	Submitting Screenshot pictures to GitHub and editing the video portion of the project	Needs everyone to submit slack messages	4 hours	2/5/20	100%
Tek Acharya	tacharya2@student.gsu.edu	Making sure we all understand our Teamwork Basics, and creating graphics for use in the video	None, we talked over it in a group meeting	4 hours	2/5/20	100%
Vedansi Parsana	vparsana1@student.gsu.edu	Programming with ARM assembly language	None	5 hours	2/5/20	100%
Zoe Kosmicki (coordinator)	zkosmicki1@student.gsu.edu	Writing the written report and turning it all in	Needs every screenshot to be uploaded to GitHub to be included in appendix	5 hours	2/6/20	Everyone in the group was incredibly responsible and communicative. Everything got done on a good time table, and everyone in the group deserves a 100%.