

Developing Soft and Parallel Programming Skills using Project-Based Learning

Spring 2020

The Commuters

Alaya Shack, Miguel Romo, Arteen Ghafourikia, Andre Nguyenphuc, Joan Galicia

**Planning and Scheduling:**

Assignee Name	Email	Task	Duration (hours)	Dependency	Due Date	Note
Alaya Shack	ashack1@student.gsu.edu	Formulate the answers to the Teamwork Basics Document	2 hours	none	02/04	Review the teamwork document of all the team members. Proofread and provide corrections if needed. Discuss questions and answers as a group.
Miguel Romo	mromo1@student.gsu.edu	Planning and scheduling as described in the assignment	30 mins	none	02/04	Update schedule as needed with duration of assignments and include notes of what members need to do and their participation.
Arteen Ghafourikia (Coordinator)	aghafourikia1@student.gsu.edu	Create Github, connect Raspberry Pi to Github, write assembly	2 hours	Github	02/04	Create an account and send member log-in information. Assist members

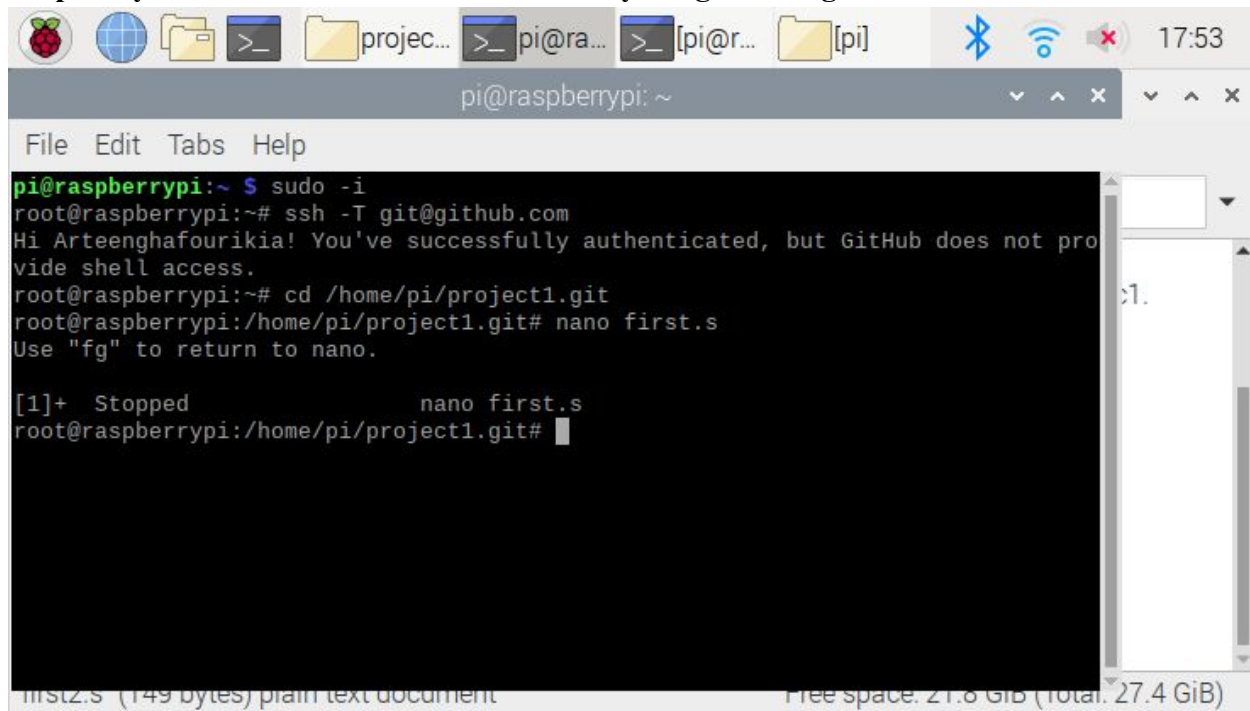
		code and edit video				with connecting Raspberry Pi to Github if needed. Edit video, and send link with log-in information for Youtube.
Joan Galicia	Jgalicia2@student.gsu.edu	Creating the slack account as described in the assignment	40 mins	Slack	02/05	Add members to slack and notify them when completed. Remind members to add their intros to slack.
Andre Nguyenphuc	anguyenphuc1@student.gsu.edu	Technical writing (getting the report ready) as described in the assignment	1 hour	Report	02/07	Review report for any corrections needed.

### **Teamwork Basics: Arteen Ghafourikia**

- **What to do to get the task accomplished and the team members' satisfaction high?**
  - It would be best if you made sure everyone is comfortable with what they are doing and make sure that people ask any questions they have. People must communicate what they want to get out of the assignment so everyone can benefit from it.
- **Work Norms:**
  - **How will work be distributed?**
    - Work will be distributed based on what people want to do, and if we cannot conclude. We will distribute the job based on people's strengths and weaknesses to get the project done as efficiently as possible.
  - **Who will set deadlines?**
    - The group will set the deadlines. However, we will make sure the deadlines are reasonable for the work that is being assigned.
  - **What happens if someone doesn't follow through on his/her commitment(for example misses a deadline)?**
    - We will see if there is anything to do to help him/her finish before the primary deadline and help them learn from their mistakes, so they can get their work done sooner. If they do not try, they will get 0% from the other members for his/her effort.
  - **How will the work be reviewed?**
    - We will each check each other's work to make sure that we turn something in that we would all be proud of.
  - **What happens if people have different opinions about the quality of the work?**
    - We will listen to each person's reasoning and turn in the quality of work that we would all be proud of. We need to turn in a job that is up to everyone's standards and not only a single individual.
  - **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).**
    - We will put deadlines that we all find reasonable, and that gives us enough time to make last-minute changes if we need to do so. People make mistakes, and we need to learn from them, and setting earlier deadlines can help us catch errors.
- **Facilitator Norms:**
  - **Will you use a facilitator?**
    - Yes, we will use a facilitator.
  - **How will the facilitator be chosen?**
    - We will see who wants to be the facilitator, and if we can't come to a conclusion, we will take a vote.
  - **Will you rotate the position?**
    - Yes, the facilitator will change.
  - **What are the responsibilities of the facilitator?**

- The Facilitator's responsibility is to keep the group on track and make sure people are getting their work done.
- **Communication Norms:**
  - **When should communication take place and through what medium (e.g., do some people prefer to communicate through email while others would rather talk on the phone)?**
    - Communication should take place at all times to make sure everyone is updated on what is going on and what needs to get done. This is a group project where everyone depends on each other to get a good grade; therefore, everyone is responsible. It can take place through any medium, but we will choose a medium that works best for everyone.
- **As a team selects two cases out of the four mentioned in handling difficult behavior. (use your own words and your own context)**
  - Too quiet- If someone is shy or too quiet, we will do our best to make them comfortable and make it a friendly and secure environment to talk in.
  - Argues-If what the person is arguing is constructive, we will use it. If he is being aggressive and mean, we will tell him/her to be less aggressive and that it is causing trouble for the team.
- **When making decisions. If the team is having trouble reaching consensus, what should you do? (use your own words and your own context)**
  - You should look at all the different options you can take logically, and as a team, you should determine which decision would be the best one to make.
- **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?**
  - We will wait and get input from everyone before we make a rash decision to increase our probability of success.
- **What happens if most people on a team want to get an "A" on the assignment, but another person decides that a "B" will be acceptable?**
  - We will discuss why the person would instead get a "B" than an "A." If the person decides not to change his/her mind, we will tell him/her that it will bring the team down, and if that does not work, we will get a third party to help us conclude.

## Raspberry PI Installation and ARM Assembly Programming: Arteen Ghafourikia



The screenshot shows a terminal window on a Raspberry Pi. The window has a title bar with icons for Raspberry Pi, a globe, a folder, and a terminal. The terminal content is as follows:

```
pi@raspberrypi:~ $ sudo -i
root@raspberrypi:~# ssh -T git@github.com
Hi Arteenghafourikia! You've successfully authenticated, but GitHub does not provide shell access.
root@raspberrypi:~# cd /home/pi/project1.git
root@raspberrypi:/home/pi/project1.git# nano first.s
Use "fg" to return to nano.

[1]+  Stopped                  nano first.s
root@raspberrypi:/home/pi/project1.git#
```

The terminal window also shows a status bar at the bottom with the text "first2.s (149 bytes) plain text document" and "Free space: 21.8 GiB (total: 27.4 GiB)".

Over here I connected to the GitHub with SSH.

```

GNU nano 3.2 first.s
@first program
.section .data
.section .text
.globl _start
_start:
mov r1, #5 @loadr1 with 5
sub r1,r1,#1 @subtract 1 from r1
add r1, r1, #4 @add 4 to r1

mov r7, #1 @ program termination: exit syscall
svc #0 @progam 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
first.s (149 bytes) plain text document Free space: 21.8 GiB (total: 27.4 GiB)

```

This is the code for the “first.s” program.

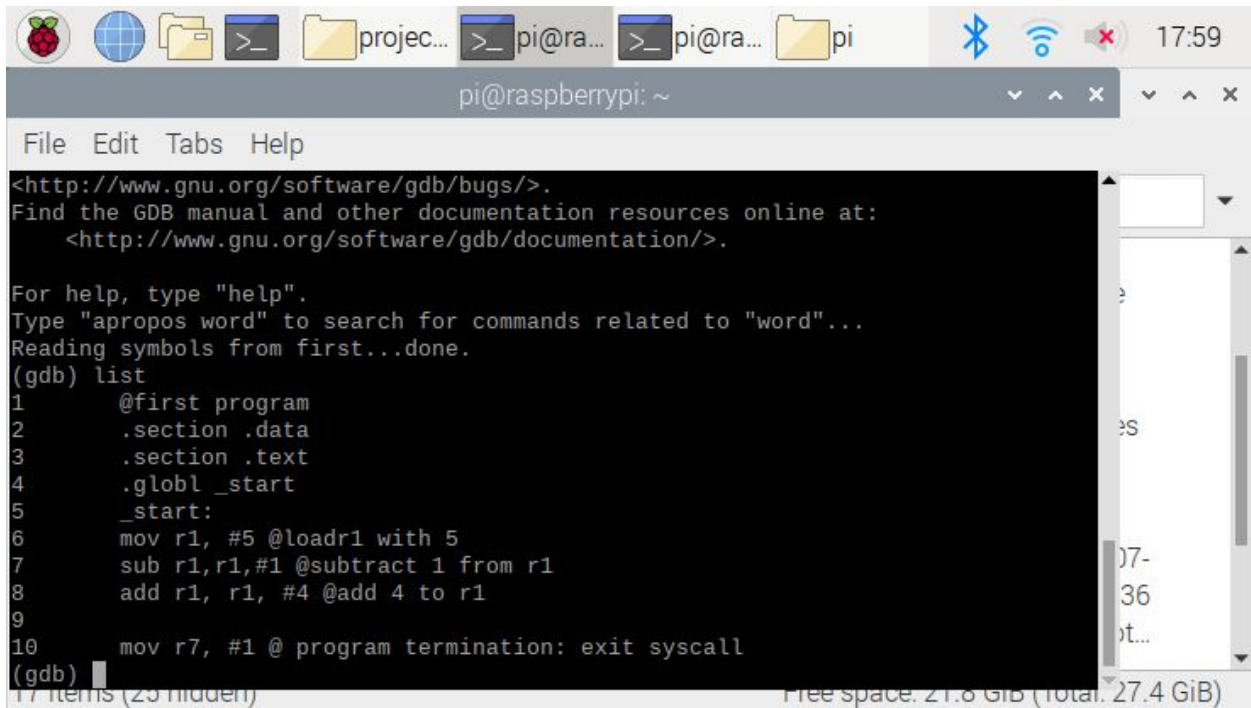
```

root@raspberrypi:/home/pi/project1.git# ld -o first first.o
root@raspberrypi:/home/pi/project1.git# 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>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "arm-linux-gnueabi".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<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 first...done.
(gdb)

```

Over here, I assembled the file, created an objective file, and linked to create an executable file. I then launched the debugger.



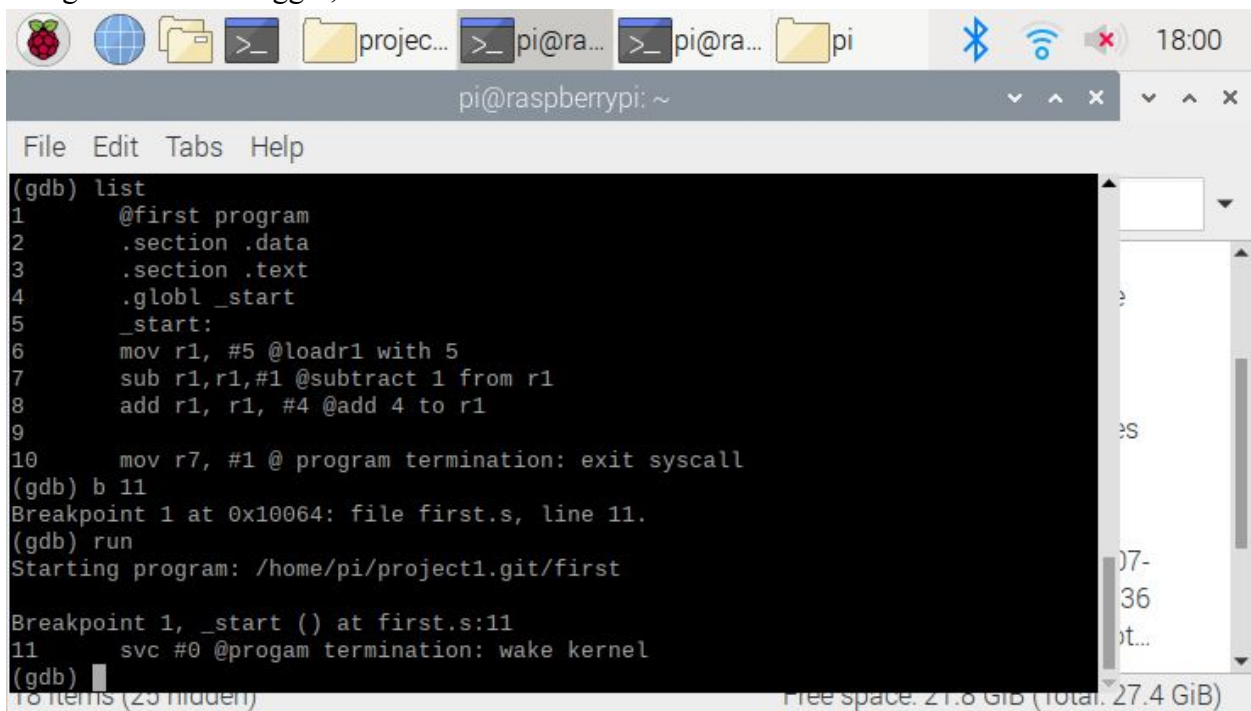
```

pi@raspberrypi: ~
File Edit Tabs Help
<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 first...done.
(gdb) list
1      @first program
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6      mov r1, #5 @loadr1 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)

```

Using the GNU debugger, I listed the first 10 lines of code from “first.s”.



```

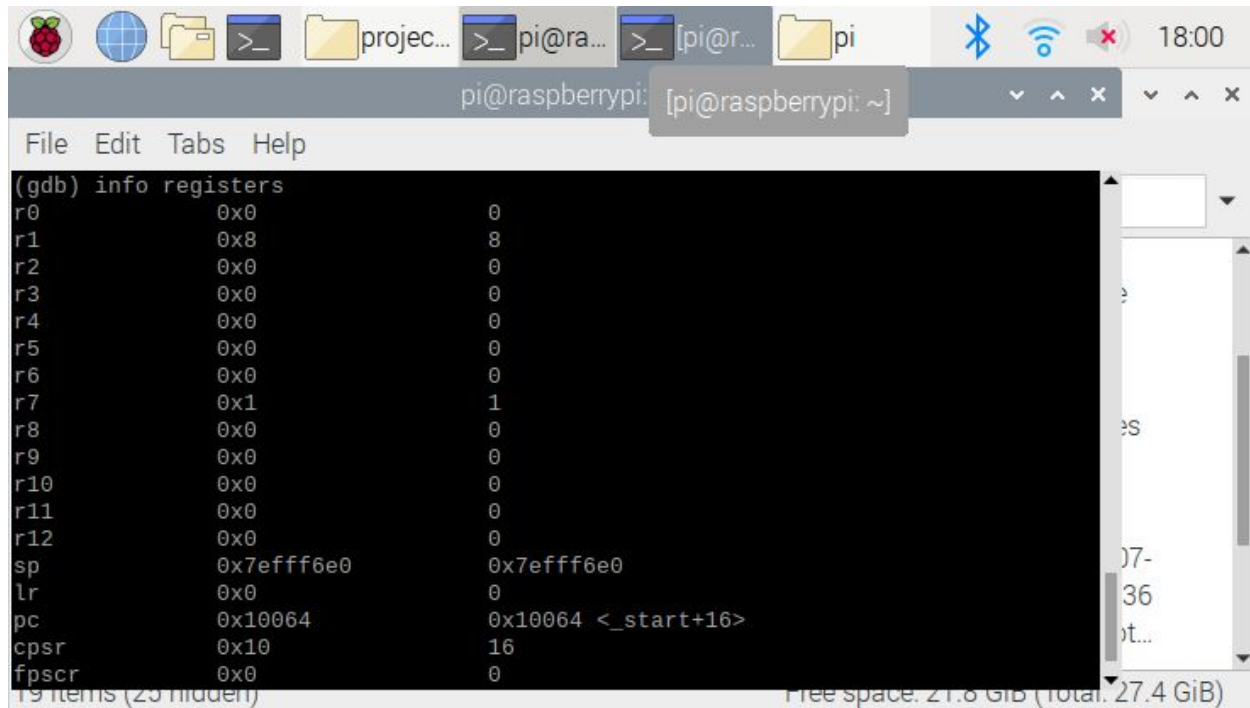
pi@raspberrypi: ~
File Edit Tabs Help
(gdb) list
1      @first program
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6      mov r1, #5 @loadr1 with 5
7      sub r1,r1,#1 @subtract 1 from r1
8      add r1, r1, #4 @add 4 to r1
9
10     mov r7, #1 @ program termination: exit syscall
(gdb) b 11
Breakpoint 1 at 0x10064: file first.s, line 11.
(gdb) run
Starting program: /home/pi/project1.git/first

Breakpoint 1, _start () at first.s:11
11     svc #0 @progam termination: wake kernel
(gdb)

```

In this screenshot, I placed a breakpoint at line 11 and then ran the program.



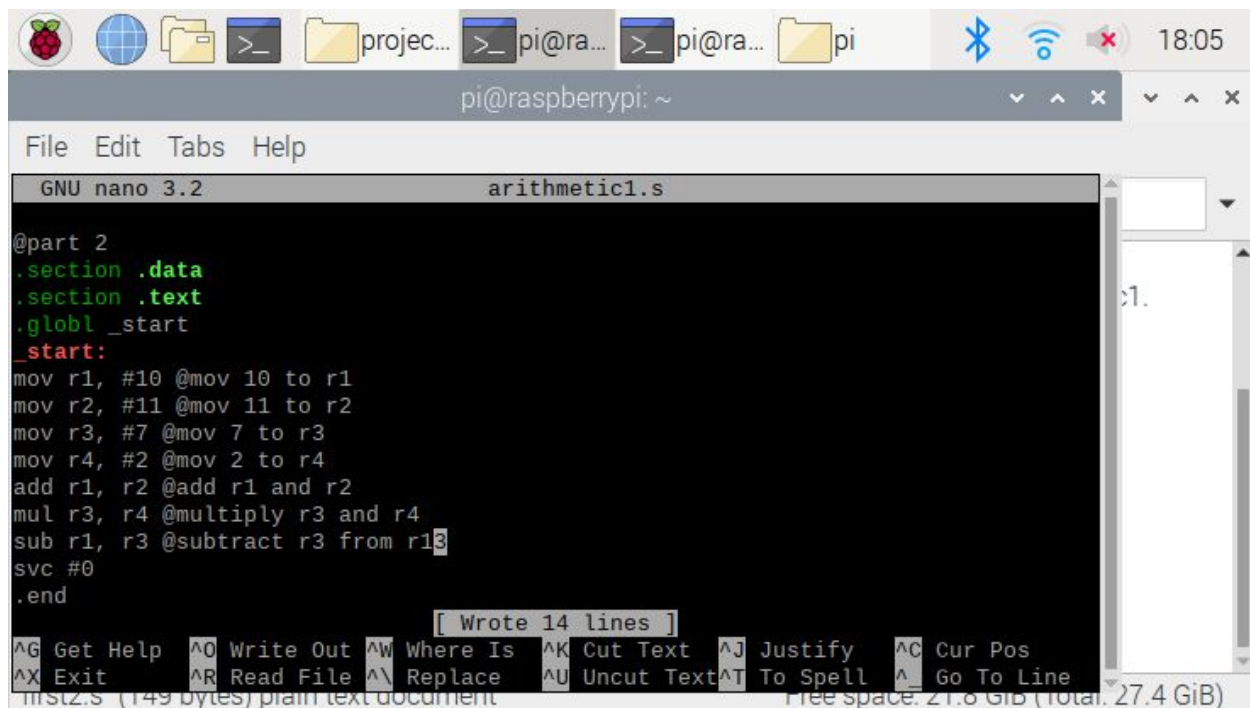


```

(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          0x7efff6e0    0x7efff6e0
lr          0x0          0
pc          0x10064      0x10064 <_start+16>
cpsr        0x10        16
fpscr       0x0          0

```

In this screenshot, I displayed the information in the registers. As you can see here in register 1 you have 8 as the value. The way this worked is that r1 had 5 in it. I then subtracted 1 and then added 4 to it which gives you the value 8. Register 7 also has 1 in it because the program moved 1 to Register 7.

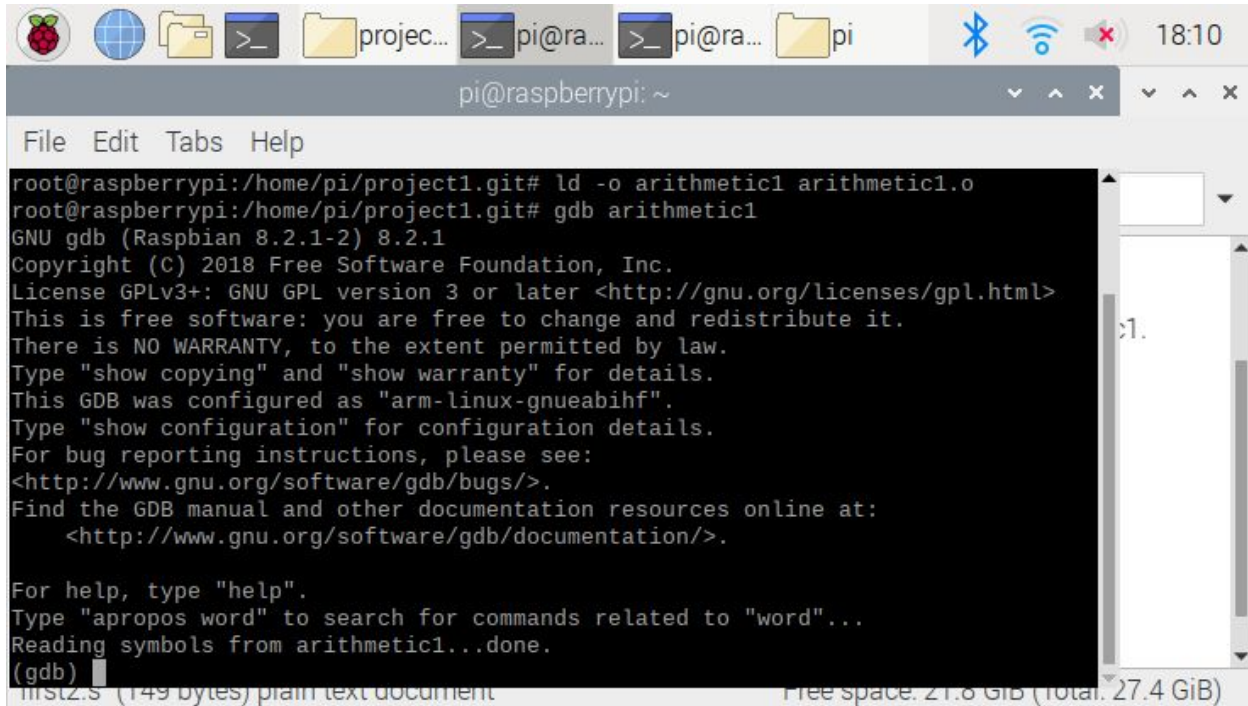


```

GNU nano 3.2 arithmetic1.s
@part 2
.section .data
.section .text
.globl _start
_start:
mov r1, #10 @mov 10 to r1
mov r2, #11 @mov 11 to r2
mov r3, #7 @mov 7 to r3
mov r4, #2 @mov 2 to r4
add r1, r2 @add r1 and r2
mul r3, r4 @multiply r3 and r4
sub r1, r3 @subtract r3 from r1
svc #0
.end

```

In part 2, this is the code for “arithmetic1.s”.



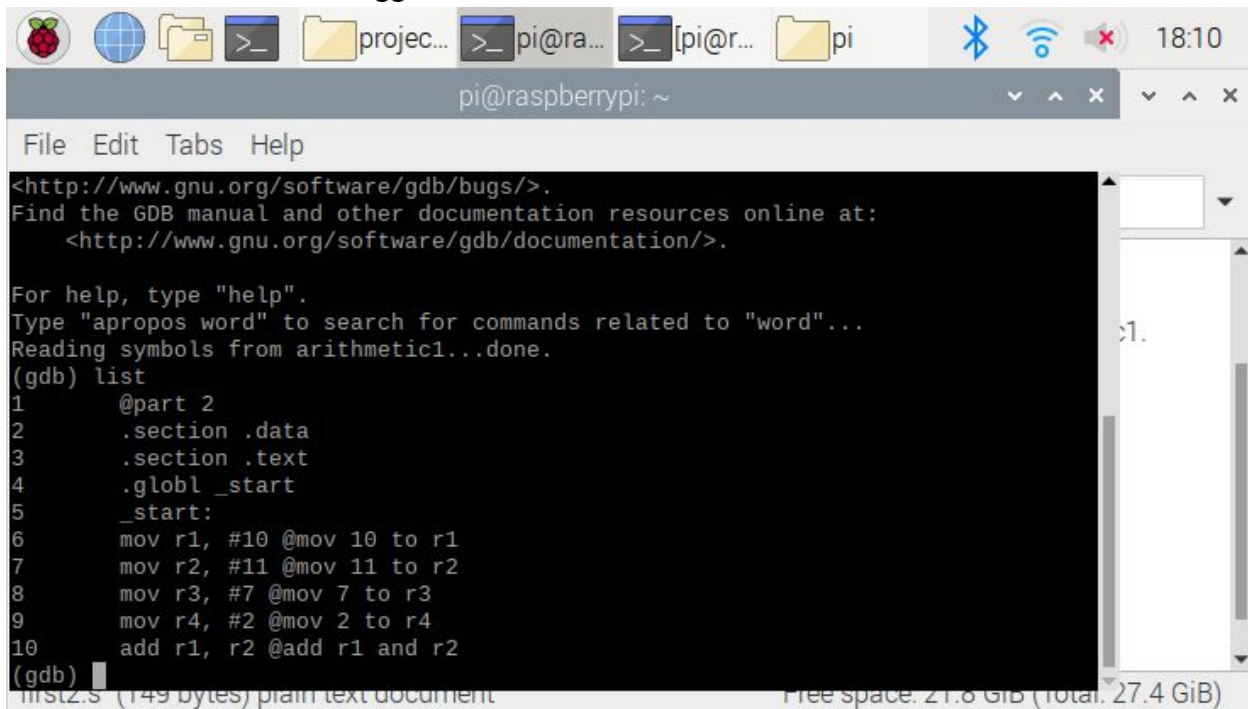
```

root@raspberrypi:/home/pi/project1.git# ld -o arithmetic1 arithmetic1.o
root@raspberrypi:/home/pi/project1.git# 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>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "arm-linux-gnueabi".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<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)

```

Over here, I assembled the file, created an objective file, and then linked to create executable file. I then launched the debugger.



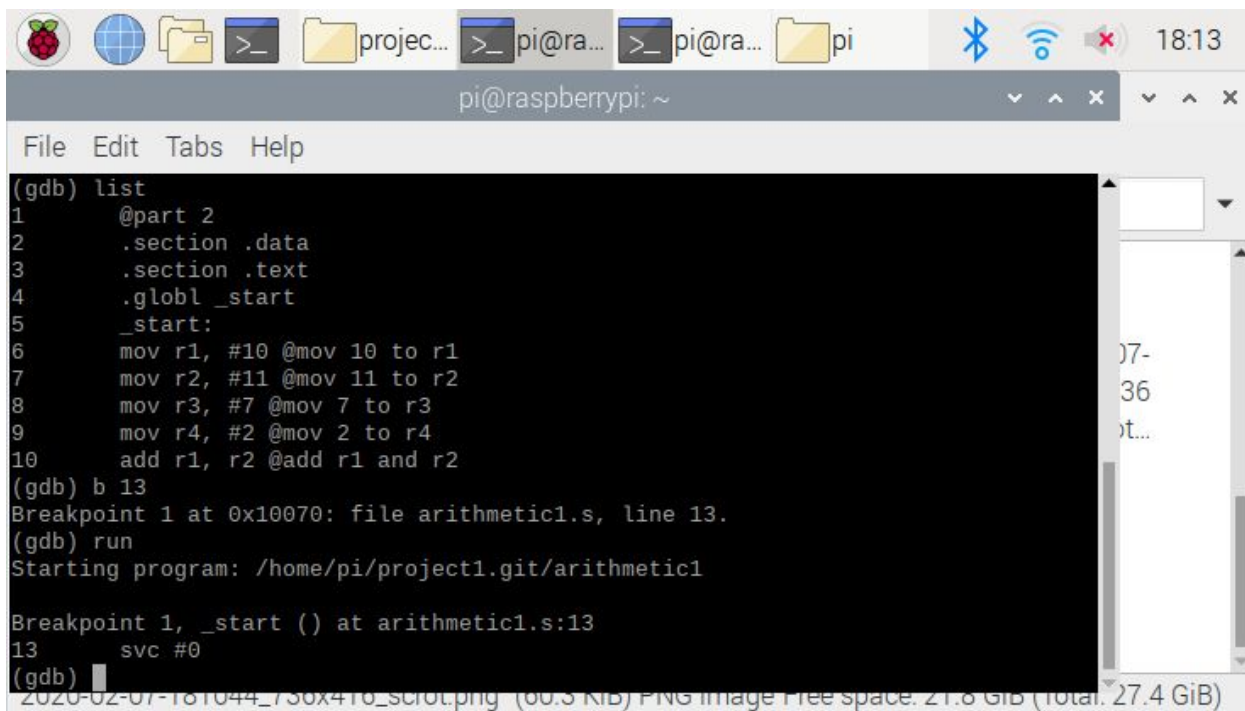
```

<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      @part 2
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6      mov r1, #10 @mov 10 to r1
7      mov r2, #11 @mov 11 to r2
8      mov r3, #7 @mov 7 to r3
9      mov r4, #2 @mov 2 to r4
10     add r1, r2 @add r1 and r2
(gdb)

```

Using the GNU debugger, I listed the first 10 lines of code from “arithmetic1.s”.



```

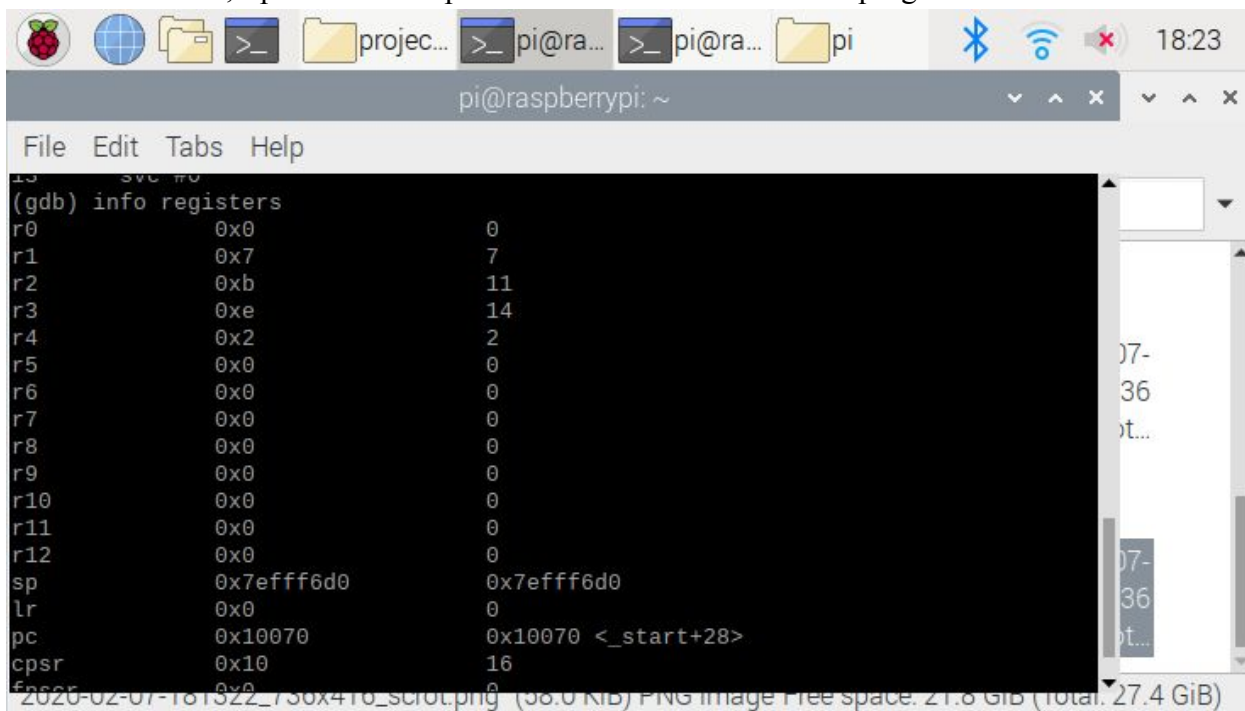
(gdb) list
1      @part 2
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6      mov r1, #10 @mov 10 to r1
7      mov r2, #11 @mov 11 to r2
8      mov r3, #7 @mov 7 to r3
9      mov r4, #2 @mov 2 to r4
10     add r1, r2 @add r1 and r2
(gdb) b 13
Breakpoint 1 at 0x10070: file arithmetic1.s, line 13.
(gdb) run
Starting program: /home/pi/project1.git/arithmetic1

Breakpoint 1, _start () at arithmetic1.s:13
13     svc #0
(gdb)

```

2020-02-07-181044\_736x416\_sshot.png (60.3 KiB) PNG image Free space: 27.8 GiB (total: 27.4 GiB)

In this screenshot, I placed a breakpoint at line 13 and then ran the program.



```

13     svc #0
(gdb) info registers
r0          0x0          0
r1          0x7          7
r2          0xb          11
r3          0xe          14
r4          0x2          2
r5          0x0          0
r6          0x0          0
r7          0x0          0
r8          0x0          0
r9          0x0          0
r10         0x0          0
r11         0x0          0
r12         0x0          0
sp          0x7efff6d0    0x7efff6d0
lr          0x0          0
pc          0x10070      0x10070 <_start+28>
cpsr       0x10        16

```

2020-02-07-181322\_736x416\_sshot.png (56.0 KiB) PNG image Free space: 27.8 GiB (total: 27.4 GiB)

```

pi@raspberrypi: ~
File Edit Tabs Help
r0      0x0      0
r1      0x7      7
r2      0xb     11
r3      0xe     14
r4      0x2      2
r5      0x0      0
r6      0x0      0
r7      0x0      0
r8      0x0      0
r9      0x0      0
r10     0x0      0
r11     0x0      0
r12     0x0      0
sp      0x7efff6d0 0x7efff6d0
lr      0x0      0
pc      0x10070 0x10070 <_start+28>
cpsr    0x10     16
fpscr   0x0      0
(gdb)

```

In these screenshots, I displayed the information in the registers. The goal of part 2 was to solve  $A=(A+B)-(C*D)$ .  $A=10$ ,  $B=11$ ,  $C=7$ ,  $D=2$ . In the code, I loaded one of the values in each register. I then added, multiplied, and subtracted accordingly which left me with 7 in Register 1 (A).  $(21)-(14)=7$ . I observed that each register that was used as a source retained its value. Only Register 3 and Register 1 changed in value because they were the only ones where I changed the value in the destination.

### Teamwork Basics: Alaya Shack

- **What to do to get the task accomplished and the team members' satisfaction high?**
  - In order to get the task accomplished and the team members' satisfaction high, the members of the group should become acquainted with each other and everyone's strengths, ground rules should be set, a facilitator should be used, lines of communication should be kept open, and we should know how to avoid or solve common problems associated with collaborative work.
- **Work Norms:**
  - **How will work be distributed?**
    - Work will be distributed evenly amongst group members. Each task will have a primary person assigned and a secondary person, in case a person is not able to follow through with their commitment and to help with reviewing and clarifying questions. We will try to assign work based on everyone's strength. Also, we will ensure that the work is rotated so that everyone will have a chance to do different types of tasks.

- **Who will set deadlines?**
  - Each member will state their idea of a reasonable due date for each task, and we will discuss and vote on a particular date.
- **What happens if someone doesn't follow through on his/her commitment?**
  - If someone doesn't follow through on his/her commitment, we will discuss why they were not able to complete their commitment. Then, the secondary person on the assigned task and the remaining group members will work to complete the task. The person who does not follow through on their commitment will receive a 0% for their effort.
- **How will work be reviewed?**
  - First, each task will be reviewed by the secondary person. Then, as a group, we will review the tasks at our meetings.
- **What happens if people have different opinions about the quality of work?**
  - If people have different opinions about the quality of work, we will discuss the issues that people have and decide what may need to be critiqued. In the end, we want the best quality, and we want success for the group.
- **What happens if people have different work habits?**
  - As long as each member gets their task accomplished by the assigned due date, the various work habits will have little to no significance. However, if a person likes to get things done early or hinders/has a negative impact on the group, we will discuss alternatives or solutions to the problem. If a person that likes to procrastinate hinders the group, we will discuss how their behavior negatively impacts the group and discuss solutions to the problem.
- **Facilitator Norms:**
  - **Will you use a facilitator?**
    - We will use a facilitator.
  - **How will the facilitator be chosen?**
    - The facilitator will be chosen based on who volunteers for the position. If no one volunteers, the facilitator will be chosen through several rounds of rock paper scissors. Once a person has held the facilitator position, they will not be eligible for rock paper scissors or to volunteer for the position.
  - **Will you rotate the position?**
    - The facilitator position will be rotated.
  - **What are the responsibilities of the facilitator?**
    - The facilitator is responsible for initiating the discussion, setting the agenda, keeping the team focused, making sure the team is progressing, ensuring every member is engaged, solving/ mollifying problems, summarizing the team's goals and decisions, and establishing a consensus among the group.
- **Communication Norms:**
  - **When should communication take place and through what medium?**

- Communication will primarily take place through text in our “groupme,” and we will communicate at our meetings, which will be held at least once a week. If more meetings are needed, we will add them to our schedule. We will communicate when we have made updates to tasks or amended tasks and for clarity on a specific task or issue.

- **Handling Difficult Behavior:**

- Too Quiet
- Argues

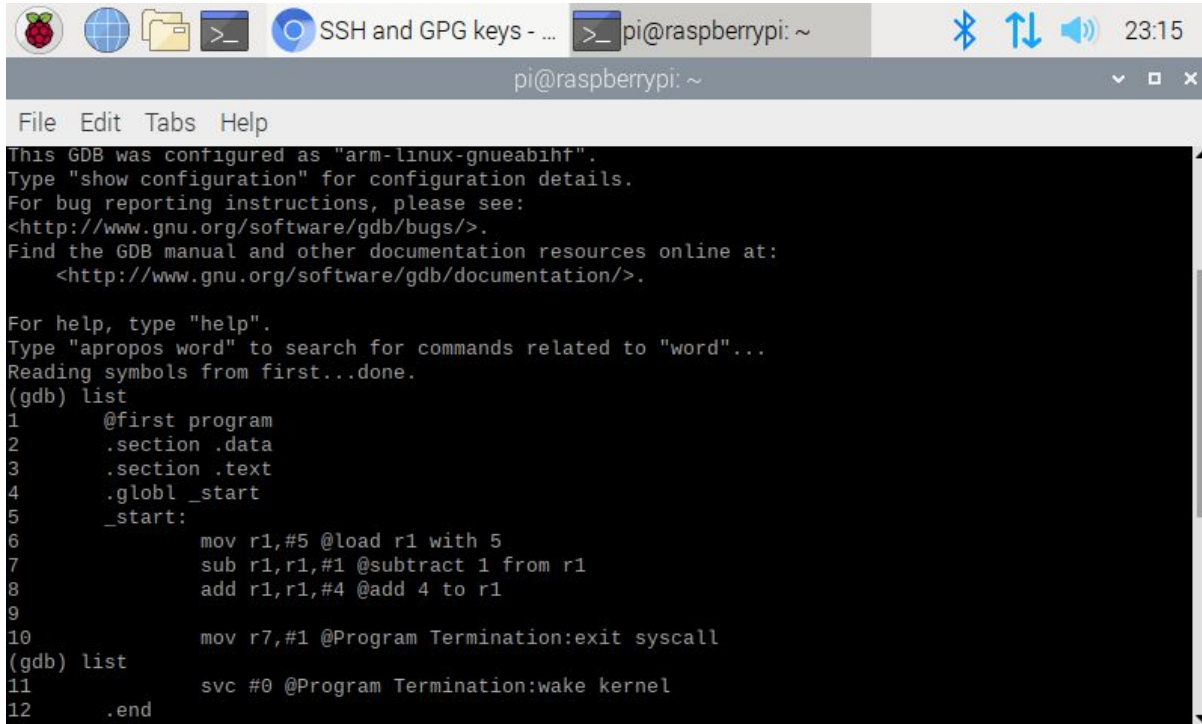
How the Person Acts	Description	What to Do
Too Quiet	This person does not actively engage in the group. They are mostly silent through discussions. They may be timid or unsure of themselves.	Make this member feel comfortable and let them know that the group values their input. Try to get them to come out of their shell by asking for their input on a specific idea or asking them a question about themselves.
Argues	This person likes to argue for fun, and they are strongly opinionated. They find somebody to argue with every opportunity the group is together. They are constantly taking up the group's time with their arguments	If the person's feedback is constructive, then we should use their feedback to ensure that we are completing quality work. However, if their feedback is negative, a discussion should be held with the facilitator so that the person will understand that their behavior is negatively impacting the group. The facilitator should let the person know that we appreciate their participation, but that their argumentative nature is not contributing to the success of the group.

- **When making decisions, If the team is having trouble reaching consensus, what should you do? (use your own words and your own context)**
  - If the team is having trouble reaching consensus, we will first consider the importance of the decision. If the decision is not significant, we will listen to each other's viewpoints and make the best decision for the overall group. If the decision is significant, we will use multivoting. The process of multivoting will consist of gathering all the ideas that we have generated. Then, we will have each member choose their top four ideas. Once the top four ideas are selected by each member, we will see which ideas were voted for the most. Next, we will identify the similarities, differences, positive aspects, and negative aspects between the ideas that were voted for the most. Now, each member will vote for their top two choices, and we will review the votes to see which choice had the most support.

- **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?**
  - If a person reaches a decision more quickly than others and pressures people to move on, ideally, the facilitator would enlist everyone's viewpoint on the decision and check to see if there is a consensus among the group. Also, the facilitator will make sure that the group has completed prior tasks that are needed before advancing to another task.
- **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 people on the team want to get an "A", but another person decides that a "B" will be acceptable, each member will share their viewpoints on the grade that they would like to receive and why they would like to receive that particular grade. Then, if the "B" team member outlook does not change, then we will continue to communicate about the issue if their behavior is a hindrance to the group, or the remaining members will find other solutions to the issue such as assigning simpler tasks to that group member or work together to raise that group member's quality of work.



## Raspberry PI Installation and ARM Assembly Programming: Alaya Shack

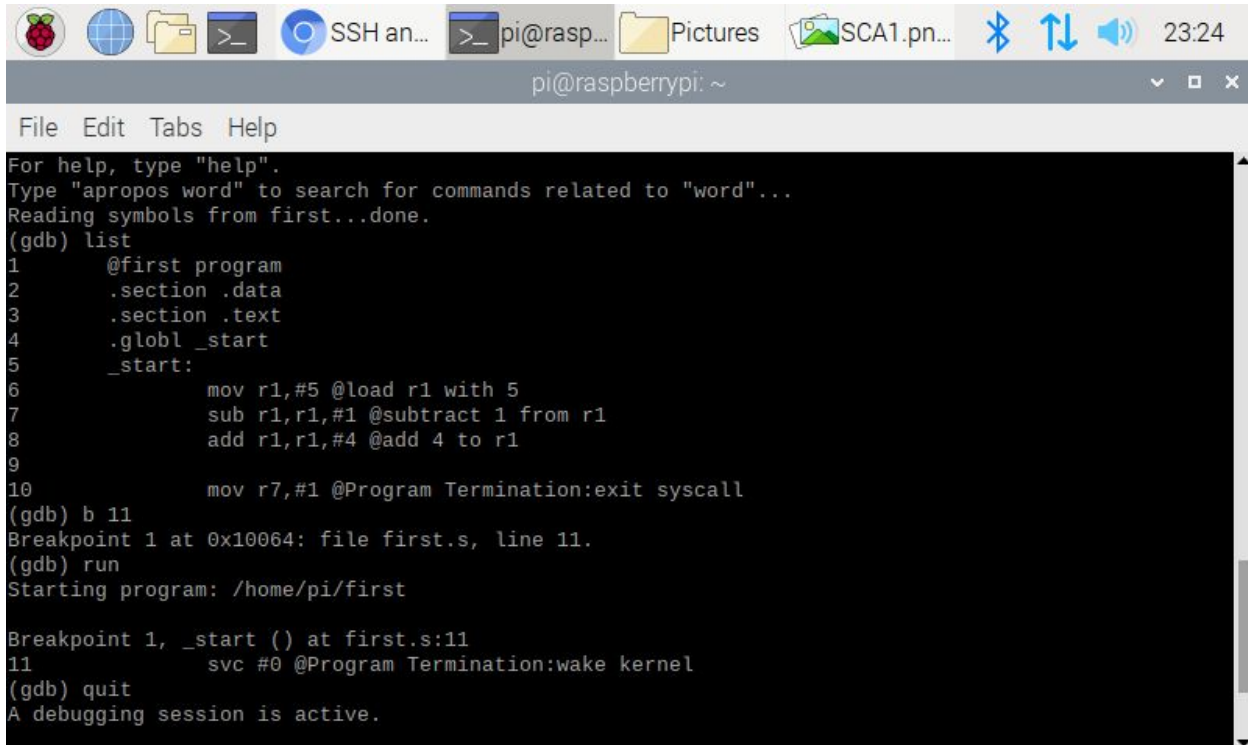
A screenshot of a terminal window on a Raspberry Pi. The window title bar shows icons for a Raspberry Pi, a globe, a folder, a terminal, and an SSH connection. The title text is "SSH and GPG keys - ... pi@raspberrypi: ~". The terminal content shows GDB startup messages and assembly code listings. The first listing shows lines 1-10 of the first program, and the second listing shows lines 11-12. The code includes section directives, global symbols, and ARM assembly instructions with comments.

```
File Edit Tabs Help
This GDB was configured as "arm-linux-gnueabi".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<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 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) list
11         svc #0 @Program Termination:wake kernel
12     .end
```

This screenshot was after I typed in the command prompt “(gdb) first”, and then I typed “(gdb) list” so that the debugger would list the first ten lines of code. Then, I typed “(gdb) list” again so that the next ten lines of code would display. This screenshot was for the “first” program.





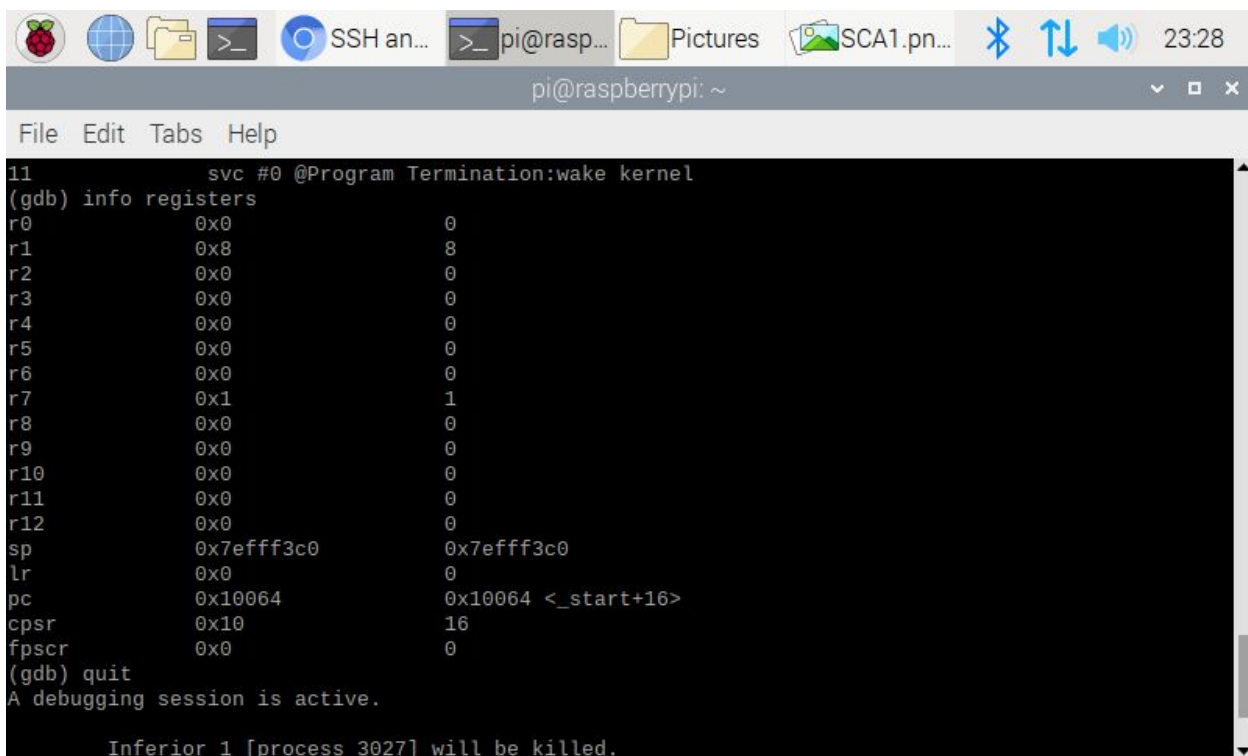
```

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 @Program Termination:exit syscall
(gdb) b 11
Breakpoint 1 at 0x10064: file first.s, line 11.
(gdb) run
Starting program: /home/pi/first

Breakpoint 1, _start () at first.s:11
11         svc #0 @Program Termination:wake kernel
(gdb) quit
A debugging session is active.

```

The second screenshot is when I inserted a breakpoint at line 11 of the first program. Next, I typed “(gdb run)” to actually run the program, and it also displays line 11 of the program. Then, I typed “(gdb quit)” to stop the debugging process.



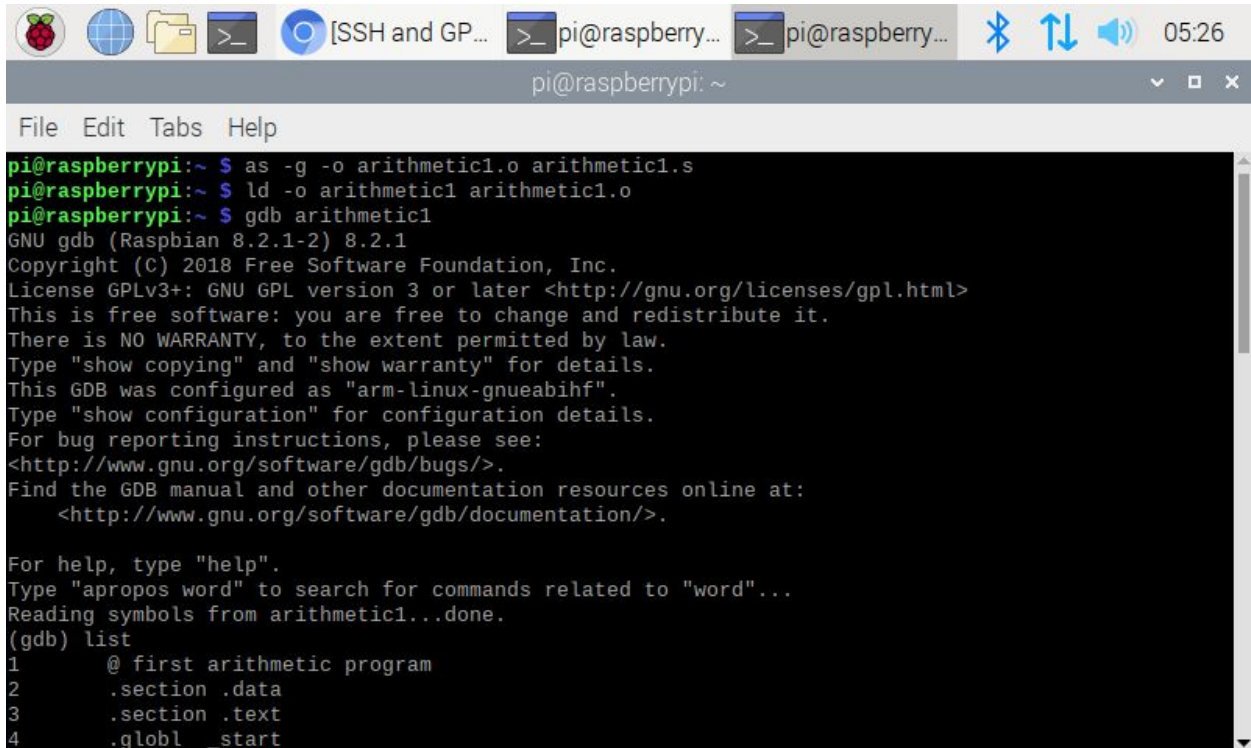
```

pi@raspberrypi: ~
File Edit Tabs Help
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) quit
A debugging session is active.

Inferior 1 [process 3027] will be killed.

```

The third screenshot displays the info registers after I typed the “(gdb) info registers” into the command line prompt. Register one has the value 8 because of the instruction `add r1,r1, #4`, which is adding the value that is already in `r1(4)` and the value 4 together, and then storing that result into `r1`. Register 7 (`r7`) has the value 1 displayed because of the `move 1 into r7` instruction.



```

pi@raspberrypi:~ $ as -g -o arithmetic1.o arithmetic1.s
pi@raspberrypi:~ $ ld -o arithmetic1 arithmetic1.o
pi@raspberrypi:~ $ 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>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "arm-linux-gnueabi".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<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      @ first arithmetic program
2      .section .data
3      .section .text
4      .globl _start

```

The fourth screenshot is of the instructions so that the first arithmetic program will be assembled, and then linked to the file to get an executable. The third line is typing the debugging command for my `arithmetic1` program.

```

File Edit Tabs Help
Reading symbols from arithmetic1...done.
(gdb) list
1      @ first arithmetic program
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6          mov r1,#10 @load r1 with 10
7          mov r2,#11 @load r2 with 11
8          add r1,r2,r1 @add r2 and r1, store in r1
9          mov r3,#7 @load r3 with 7
10         mov r4,#2 @load r4 with 2
(gdb) list
11         mul r5,r3,r4 @multiply r3 with r4, store in r5
12         sub r1,r1,r5 @subtract r5 from r1, store in r1
13         svc #0 @Program Termination: wake kernel
14         .end
(gdb) b 13
Breakpoint 1 at 0x10070: file arithmetic1.s, line 13.
(gdb) run
Starting program: /home/pi/arithmetic1

Breakpoint 1, _start () at arithmetic1.s:13
13         svc #0 @Program Termination: wake kernel

```

The fifth screenshot is using the “(gdb) list” command to display the first ten lines of code. Then, the second “(gdb) list” displays the next ten lines of code/the rest of the code. In my code, I inserted comments with the “@” to explain what the instructions do. Also, I inserted the breakpoint at line 13. Then, I did “(gdb) run” to start the debugging process.

```

(gdb) b 13
Breakpoint 1 at 0x10070: file arithmetic1.s, line 13.
(gdb) run
Starting program: /home/pi/arithmetic1

Breakpoint 1, _start () at arithmetic1.s:13
13      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          0xe          14
r6          0x0          0
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>

```

The sixth screenshot is of the registers. In r1, the value is 7 because the value in r5(14) was subtracted from r1(21), which equals 7. In r2, the value is 11 because 11 was loaded into r2. In r3, the value is 7 because the value 7 was loaded into r3. In r4, the value is 2 because the value 2 was loaded into r4. In r5, the value is 14 because the values in r3(7) and r4(2) multiplied are 14, and that value had to be loaded into its own register(r5) or there would be an error message.

The screenshot shows a terminal window titled 'pi@raspberrypi: ~'. The window contains the following text:

```

Breakpoint 1, _start () at arithmetic1.s:13
13      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          0xe          14
r6          0x0          0
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) quit
A debugging session is active.

```

The seventh screenshot is of the registers, and then quitting the debugging process.

### **Teamwork Basics: Andre Nguyenphuc**

- **What to do to get the task accomplished and the team members' satisfaction high?**
  - Make sure that everybody in the group knows that their opinion is valued, and make sure that everybody in the group knows what to do through clear communication between everybody

- **Work Norms:**

- **How will work be distributed?**
  - The tasks will be distributed evenly among the group members and will also be distributed based on their strength/knowledge of the task. For example, if someone knows how to shoot and edit videos, then they will be assigned to do the video task for the project.
- **Who will set deadlines?**
  - Everyone will come together and discuss the best date deadline by suggesting dates and voting which date will work fairly for everyone.
- **What happens if someone doesn't follow through on his/her commitment?**
  - The person who did not follow through on his/her commitment will get a 0% on their contribution to the group and will have to explain why they did not follow through so the same does not happen for the next assignments
- **How will work be reviewed?**
  - As a group, everybody will review all the tasks to make sure the task is complete and correct.
- **What happens if people have different opinions about the quality of work?**
  - If people have different opinions about the quality of work then as a group we should set a bar on what is the beneficial for the group, so if everybody agrees to have a high standard it is that person's responsibility to be the same standard
- **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)?**
  - If the task is accomplished, then the work habits do not matter as long as it does not negatively affect the way other group members work on their tasks.

- **Facilitator Norms:**

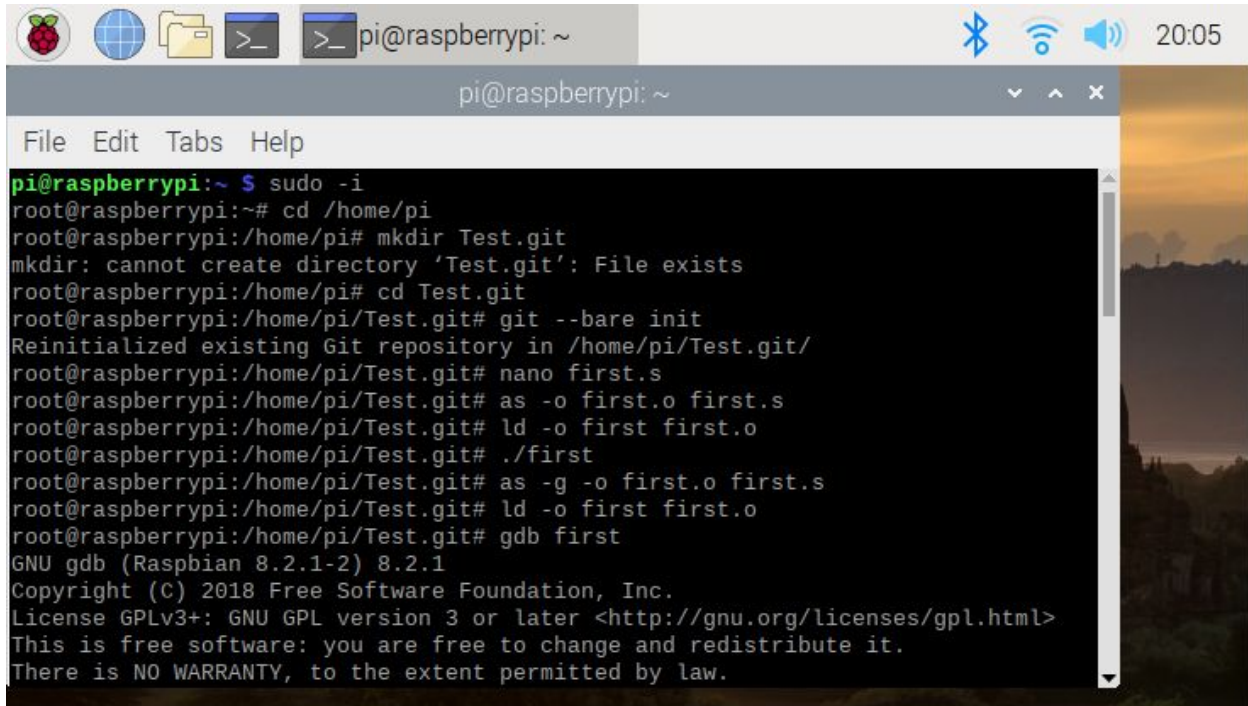
- **Will you use a facilitator?**
  - Yes, a facilitator will be used
- **How will the facilitator be chosen?**
  - The facilitator is chosen by whoever wants to be the facilitator. If there is no one who wants to be it, then we will vote.
- **Will you rotate the position?**
  - Yes, the position will be rotated.
- **What are the responsibilities of the facilitator?**
  - The responsibilities include having the team focus on the task, make sure every team member is participating in discussions, reminding all the team members on the agreed-upon time for meetings and deadlines, when something is stalled suggest alternatives, help team members confront problems, and summarize and clarify the team's decisions.

- **Communication Norms:**

- **When should communication take place and through what medium?**

- Communication will mainly take place in whichever medium the group agrees on and should always be taking place so that everyone is well informed about any questions or problems that occur.
- **As a team selects two cases out of the four mentioned in handling difficult behavior. (use your own words and your own context)**
  - Too quiet - Try to include the person in the conversation by asking them questions or getting them to participate in tasks that they know how to do
  - Argues - If what the person is saying is constructive and overall seems like it could help, then the group should consider their opinion. However, if it seems just negative, then the facilitator should inform them that their actions are only negatively impacting the group, which would decrease the quality of the group's work.
- **When making decisions, If the team is having trouble reaching consensus, what should you do? (use your own words and your own context)**
  - If the team cannot reach a consensus, we should consider everybody's opinion and do a vote to see what the group believes will benefit the project. If there is still no decision, the team coordinator should decide what is best for the project.
- **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?**
  - If they pressure other people, the facilitator should let them know how their pressure is negatively affecting the group and that rushing people to make decisions could lead to bigger problems.
- **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?**
  - We will tell the person who wants a B instead of an A how an A is better for the overall group. If they still do not change their mind, as a group try to find solutions that will please everyone, such as either giving the B person less tasks to do or deducting points from them during the evaluation.



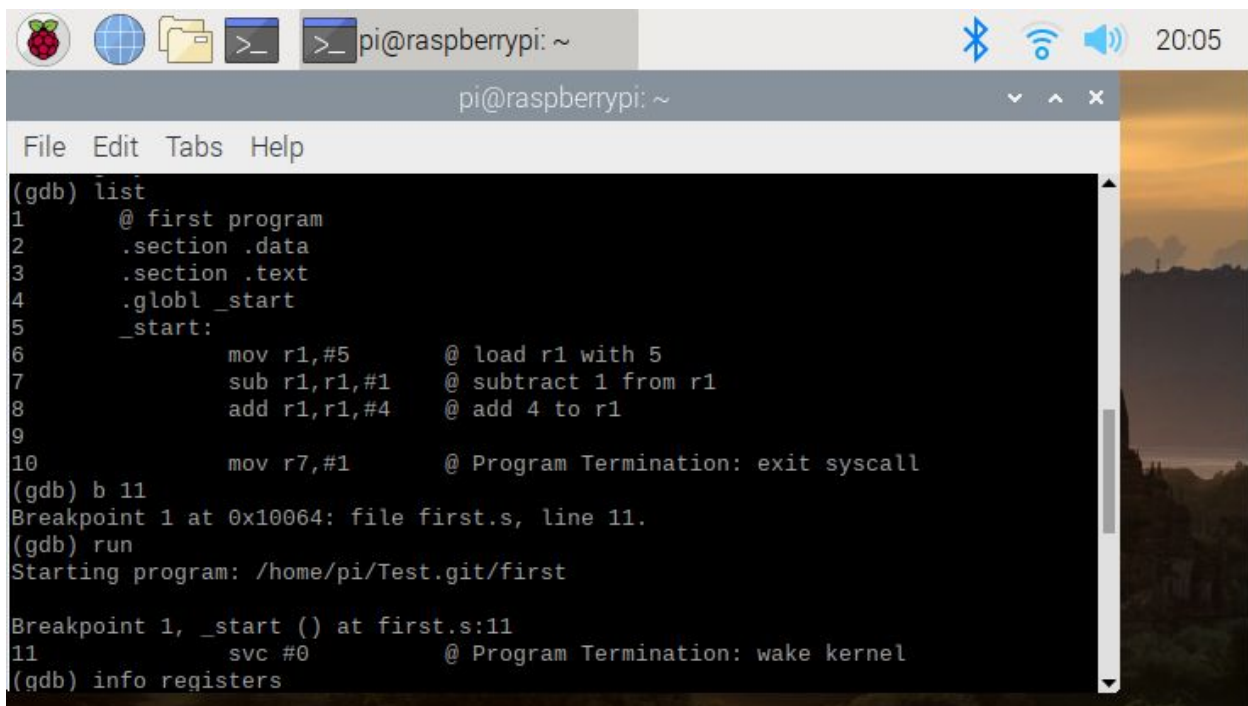


```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo -i
root@raspberrypi:~# cd /home/pi
root@raspberrypi:/home/pi# mkdir Test.git
mkdir: cannot create directory 'Test.git': File exists
root@raspberrypi:/home/pi# cd Test.git
root@raspberrypi:/home/pi/Test.git# git --bare init
Reinitialized existing Git repository in /home/pi/Test.git/
root@raspberrypi:/home/pi/Test.git# nano first.s
root@raspberrypi:/home/pi/Test.git# as -o first.o first.s
root@raspberrypi:/home/pi/Test.git# ld -o first first.o
root@raspberrypi:/home/pi/Test.git# ./first
root@raspberrypi:/home/pi/Test.git# as -g -o first.o first.s
root@raspberrypi:/home/pi/Test.git# ld -o first first.o
root@raspberrypi:/home/pi/Test.git# 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>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

```

Creation of test folder and assembling and linking of first program file. The command `./first` does not run the file because it is not debugged yet



```

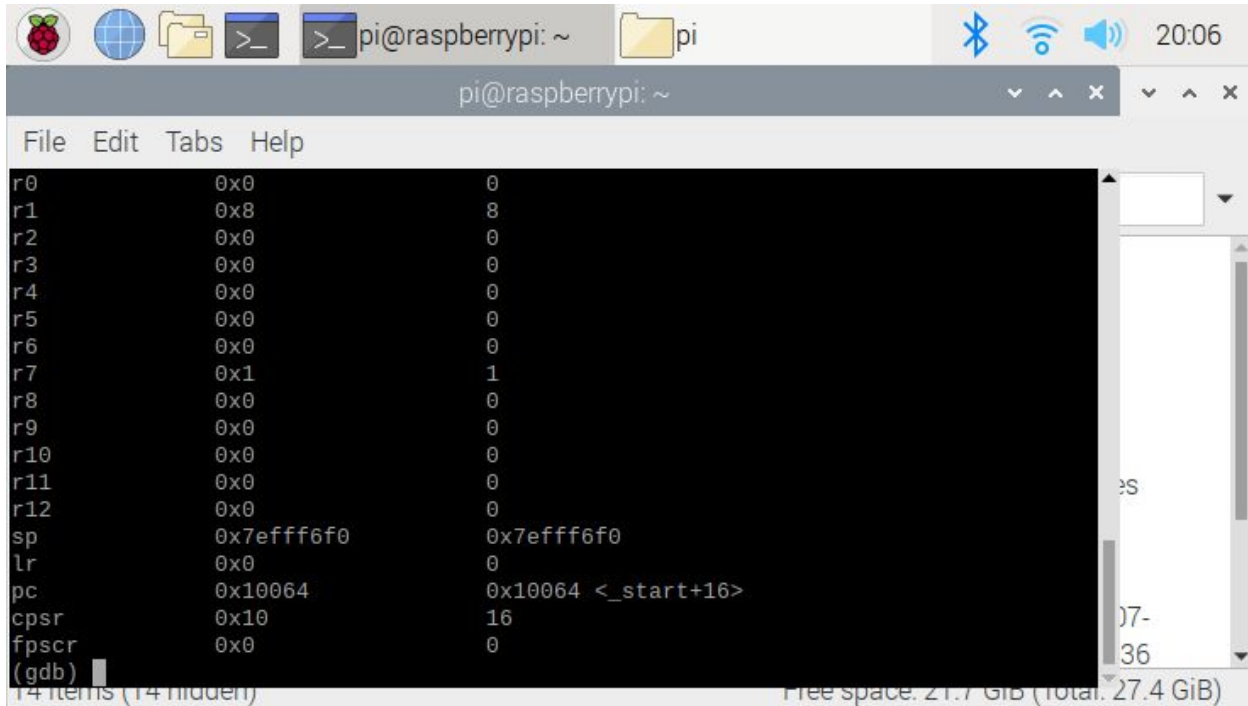
pi@raspberrypi: ~
File Edit Tabs Help
(gdb) list
1      @ first program
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6          mov r1,#5          @ load r1 with 5
7          sub r1,r1,#1       @ subtract 1 from r1
8          add r1,r1,#4       @ add 4 to r1
9
10         mov r7,#1          @ Program Termination: exit syscall
(gdb) b 11
Breakpoint 1 at 0x10064: file first.s, line 11.
(gdb) run
Starting program: /home/pi/Test.git/first

Breakpoint 1, _start () at first.s:11
11         svc #0             @ Program Termination: wake kernel
(gdb) info registers

```

Did (gdb) list to list the first 10 lines of the code. Code for the first program file had to include a breakpoint so we can examine the register or memory.



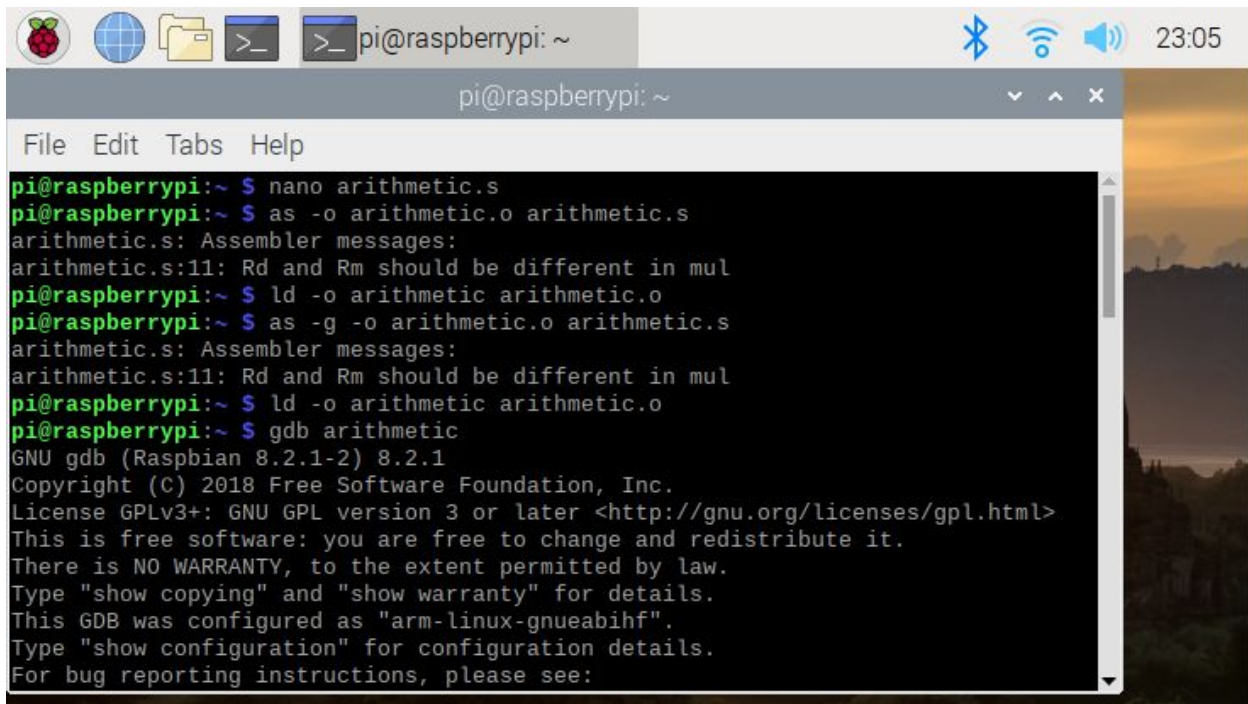


The screenshot shows a terminal window titled 'pi@raspberrypi: ~' with a menu bar (File, Edit, Tabs, Help). The terminal displays the following register values:

Register	Value	Comment
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	0x7efff6f0	0x7efff6f0
lr	0x0	0
pc	0x10064	0x10064 <_start+16>
cpsr	0x10	16
fpscr	0x0	0

The prompt is '(gdb)'. At the bottom, it says '14 items (14 hidden)' and 'Free space: 21.7 GiB (total: 27.4 GiB)'.

Info in the registers include 8 being in r1 and 1 being in r7 because we subtracted 1 from r, whose initial value was 5 and then we added 4 to r1, which makes the value in r1 8. For r7, we moved the value of 1 to r7 in line 10.



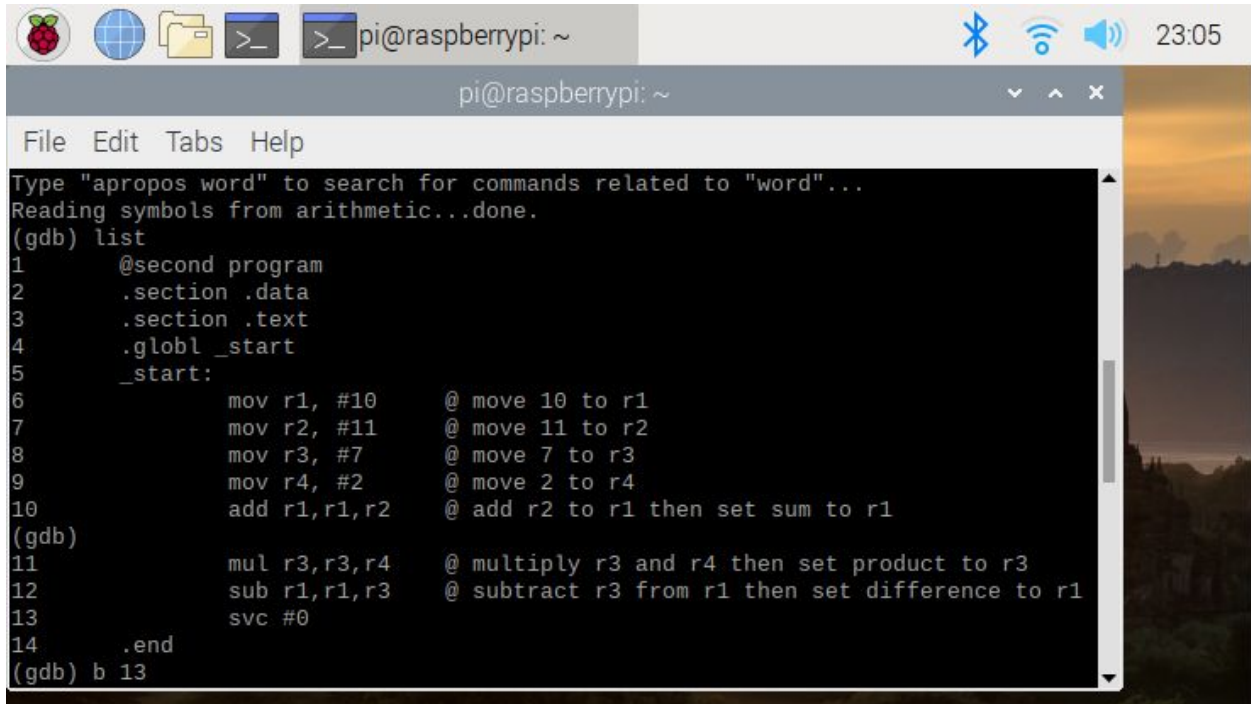
The screenshot shows a terminal window titled 'pi@raspberrypi: ~' with a menu bar (File, Edit, Tabs, Help). The terminal displays the following commands and output:

```

pi@raspberrypi:~ $ nano arithmetic.s
pi@raspberrypi:~ $ as -o arithmetic.o arithmetic.s
arithmetic.s: Assembler messages:
arithmetic.s:11: Rd and Rm should be different in mul
pi@raspberrypi:~ $ ld -o arithmetic arithmetic.o
pi@raspberrypi:~ $ as -g -o arithmetic.o arithmetic.s
arithmetic.s: Assembler messages:
arithmetic.s:11: Rd and Rm should be different in mul
pi@raspberrypi:~ $ ld -o arithmetic arithmetic.o
pi@raspberrypi:~ $ gdb arithmetic
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>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "arm-linux-gnueabi".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:

```

Assembling, linking, and doing the debugging command for the arithmetic file.



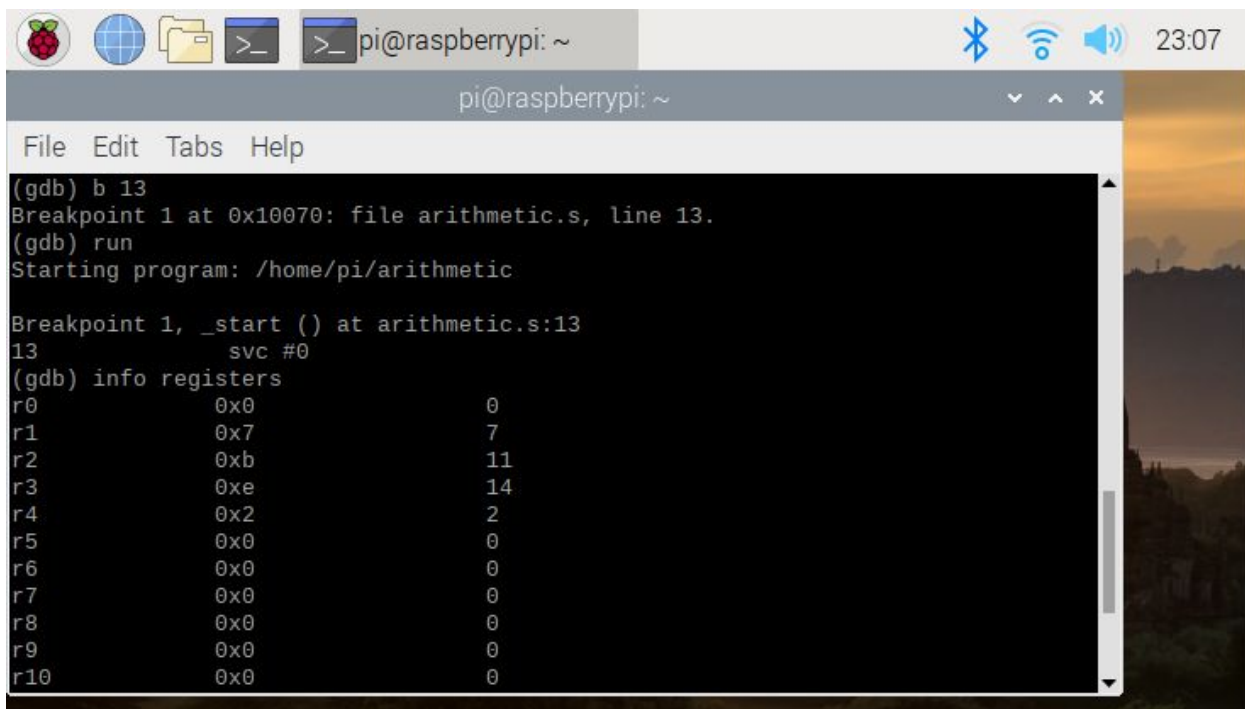
The screenshot shows a terminal window on a Raspberry Pi. The window title is "pi@raspberrypi: ~". The terminal displays the following text:

```

Type "apropos word" to search for commands related to "word"...
Reading symbols from arithmetic...done.
(gdb) list
1      @second program
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6          mov r1, #10      @ move 10 to r1
7          mov r2, #11      @ move 11 to r2
8          mov r3, #7       @ move 7 to r3
9          mov r4, #2       @ move 2 to r4
10         add r1,r1,r2      @ add r2 to r1 then set sum to r1
(gdb)
11         mul r3,r3,r4      @ multiply r3 and r4 then set product to r3
12         sub r1,r1,r3      @ subtract r3 from r1 then set difference to r1
13         svc #0
14         .end
(gdb) b 13

```

Code for the arithmetic program. Setting values to different registers then using different arithmetic to get the answer for A(r1). Also, using (gdb) list to list the first 10 lines of code, then hitting enter to list the rest of the lines of code. I first did the add command to r1 and r2, which is 21 and set that value to r1. I then multiplied r4 to r3, which made me get 14, and I set the product to r3. I then subtracted r3 from r1 (21-14) and got 7, then set the answer to r1, which is shown in the register.



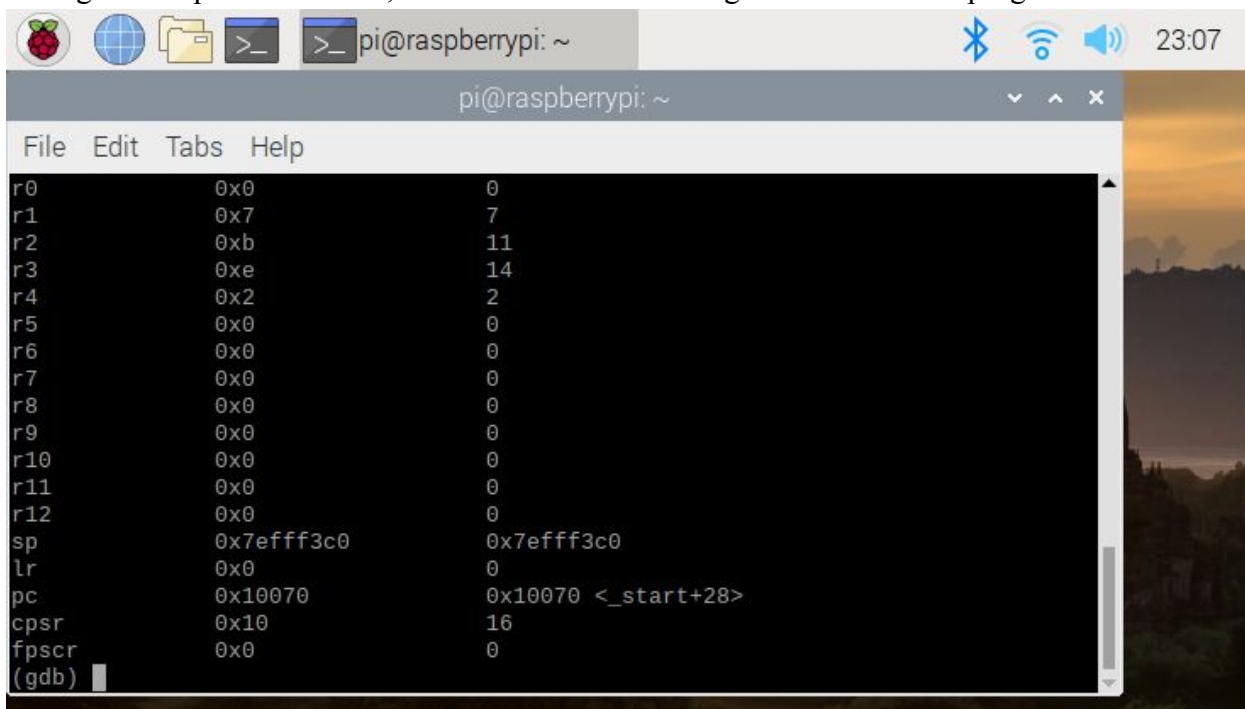
```

pi@raspberrypi: ~
(gdb) b 13
Breakpoint 1 at 0x10070: file arithmetic.s, line 13.
(gdb) run
Starting program: /home/pi/arithmetic

Breakpoint 1, _start () at arithmetic.s:13
13      svc #0
(gdb) info registers
r0          0x0          0
r1          0x7          7
r2          0xb          11
r3          0xe          14
r4          0x2          2
r5          0x0          0
r6          0x0          0
r7          0x0          0
r8          0x0          0
r9          0x0          0
r10         0x0          0

```

Setting a breakpoint at line 13, so we can examine the registers and run the program.



```

pi@raspberrypi: ~
r0          0x0          0
r1          0x7          7
r2          0xb          11
r3          0xe          14
r4          0x2          2
r5          0x0          0
r6          0x0          0
r7          0x0          0
r8          0x0          0
r9          0x0          0
r10         0x0          0
r11         0x0          0
r12         0x0          0
sp          0x7efff3c0    0x7efff3c0
lr          0x0          0
pc          0x10070      0x10070 <_start+28>
cpsr       0x10         16
fpscr      0x0          0
(gdb)

```

R1 changed because I set the total answer to r1, and r3 changed because I set the product of r3 and r4 to r3. R2 and r4 values remained unchanged because I did not set any values to them besides their initial values.

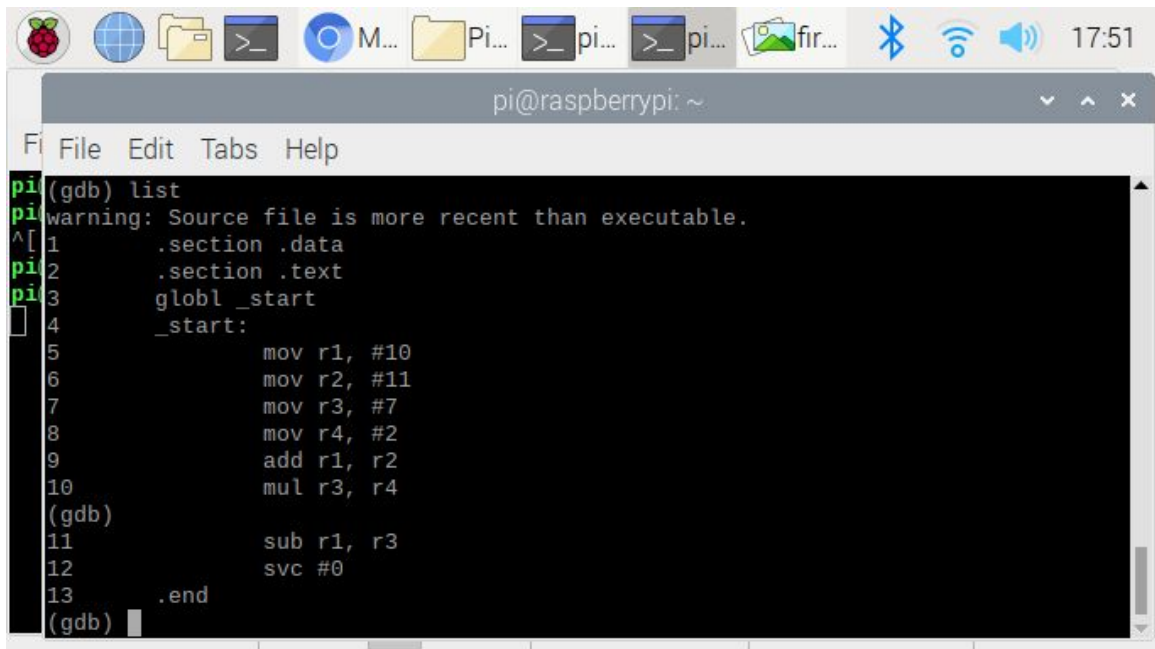
### **Teamwork Basics: Joan Galicia**

- **What to do to get the task accomplished and the team members' satisfaction high?**
  - Meet together in person and make a plan of who does what task, create a good working environment so people can ask for help, create a deadline for the people to complete their tasks.
- **Work Norms:**
  - **How will work be distributed?**
    - Work will be distributed evenly and on a volunteer basis if more than one person wants the task we collectively as a group decide who gets it based on their strengths and weaknesses.
  - **Who will set deadlines?**
    - As a group, we will all set a certain deadline for when we should have most of our work done.
  - **What happens if someone doesn't follow through on his/her commitment (for example misses a deadline)?**
    - The person should contact us in time for this type of issue because it can be resolved easily before the due date. Although if this is not the case and we are approaching a deadline, we will consult as a group and decide whether the person who cannot commit deserves the low satisfaction report.
  - **How will the work be reviewed?**
    - Each person takes a turn looking at a section of the work and writing down whether there is an issue after everyone has reviewed each section of the tasks given.
  - **What happens if people have different opinions about the quality of the work?**
    - Everyone should take into consideration the quality. and as a group, we should decide if it should be redone with assistance from everyone or convince the person with a different opinion that it is to our standard.
  - **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).**
    - Habits can be changed with time. It's only as long as the group continues to cooperate with each other that we can finish the assignment on time. If a habit is causing issues in the group, then we will all come together to set aside our differences and adapt to each other's habits.
- **Facilitator Norms:**
  - **Will you use a facilitator?**
    - Yes, we will use a facilitator.
  - **How will the facilitator be chosen?**
    - If no one volunteers, then we take a vote on someone and when another project comes around, we take turns being the facilitator.
  - **Will you rotate the position?**
    - Yes, we will rotate the position.

- **What are the responsibilities of the facilitator?**
  - This person's responsibilities are to keep the group focused by keeping us on the same agenda, and summarizing what we have done in the group so far. This person is also in the group to support everyone and in-turn we should do the same for the facilitator.
- **Communication Norms:**
  - **When should communication take place and through what medium (e.g., do some people prefer to communicate through email while others would rather talk on the phone)?**
    - Our main medium for conversation takes place through phone messaging as that is what we have decided. If we need to send some work information, we decided to do that through email and give notice to the group as a whole.
- **As a team selects two cases out of the four mentioned in handling difficult behavior. (use your own words and your own context)**
  - Too quiet - Create a friendly environment and invite this person to share their opinions on the task. Continue doing this until the person is comfortable without drawing them out.
  - Argues - A person who argues is not a bad thing as this person can be used to question what we are doing and allow us to better handle a task together. As long as other person is not affecting the group negatively there is no main issue, but if there is an issue, we as a group would talk to the person to calm them.
- **When making decisions. If the team is having trouble reaching consensus, what should you do? (use your own words and your own context)**
  - As a group we will decide everything collectively and for everyone to get a chance to voice their decision, I would pinpoint the main objective and compare everyone's decision and give the pros and cons.
- **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?**
  - If the person is still pressuring the group to move on, as a group, we will calm the person down and evaluate how negatively he/she is affecting productivity by making us speed up.
- **What happens if most people on a team want to get an “A” on the assignment, but another person decides that a “B” will be acceptable?**
  - The person who considers B to be an acceptable grade will not give his/her full effort on the project. It is best that as a group we decide to convince him to put an effort in getting an A. His/her lack of effort may just come from the difficulty of asking for help and as a group we are here are everyone.

## Teamwork Basics: Joan Galicia

### Arithmetic1

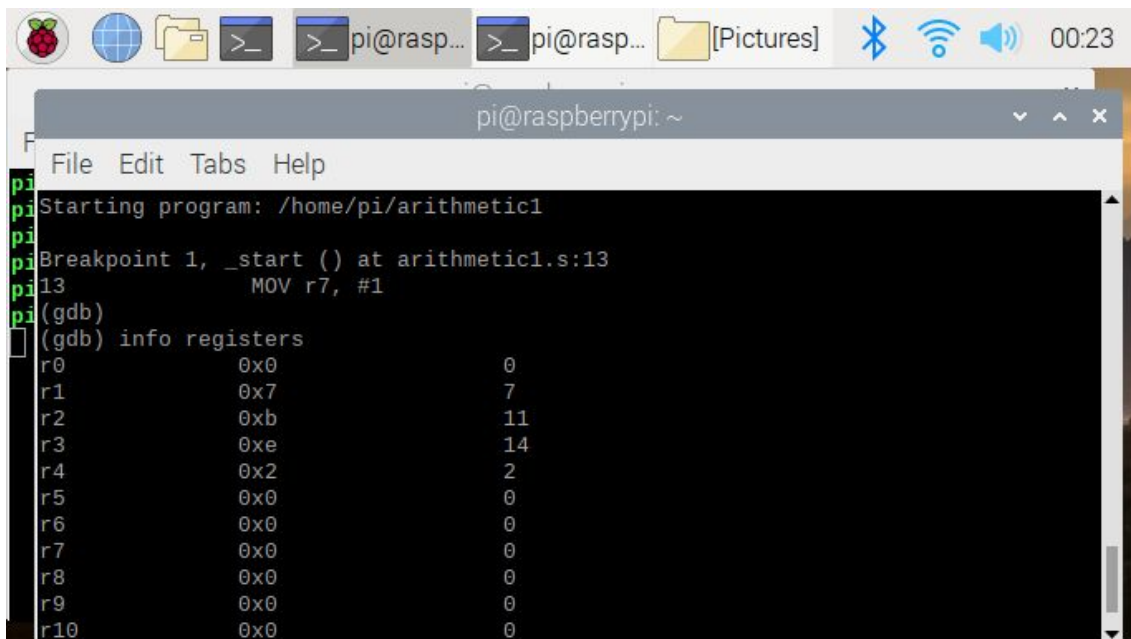


```

pi@raspberrypi: ~
File Edit Tabs Help
(gdb) list
warning: Source file is more recent than executable.
^
1      .section .data
pi 2      .section .text
pi 3      globl _start
pi 4      _start:
5          mov r1, #10
6          mov r2, #11
7          mov r3, #7
8          mov r4, #2
9          add r1, r2
10         mul r3, r4
(gdb)
11         sub r1, r3
12         svc #0
13     .end
(gdb)

```

In this screen shot, I am already in the GDB debugger and am displaying the contents of my code for the first part of the project, “first.s”.



```

pi@raspberrypi: ~
File Edit Tabs Help
Starting program: /home/pi/arithmetic1
pi
pi Breakpoint 1, _start () at arithmetic1.s:13
pi 13         MOV r7, #1
pi (gdb)
pi (gdb) info registers
r0          0x0          0
r1          0x7          7
r2          0xb         11
r3          0xe         14
r4          0x2          2
r5          0x0          0
r6          0x0          0
r7          0x0          0
r8          0x0          0
r9          0x0          0
r10         0x0          0

```

I have added my breakpoint to line 13 of my code because a breakpoint has to be placed after my last operation. Once this was done, I then typed “run” and then checked the info registers to find which register my value was stored under.



```

pi@raspberrypi: ~
File Edit Tabs Help
pi r0      0x0      0
pi r1      0x7      7
pi r2      0xb     11
pi r3      0xe     14
pi r4      0x2      2
pi r5      0x0      0
pi r6      0x0      0
pi r7      0x0      0
pi r8      0x0      0
pi r9      0x0      0
pi r10     0x0      0
pi r11     0x0      0
pi r12     0x0      0
pi sp      0x7efff3b0  0x7efff3b0
pi lr      0x0      0
pi pc      0x10070   0x10070 <_start+28>
--Type <RET> for more, q to quit, c to continue without paging--

```

This shows the rest of my info registers and the answer to the prompt given in the project was to perform an arithmetic operation where the value is stored in register 1.

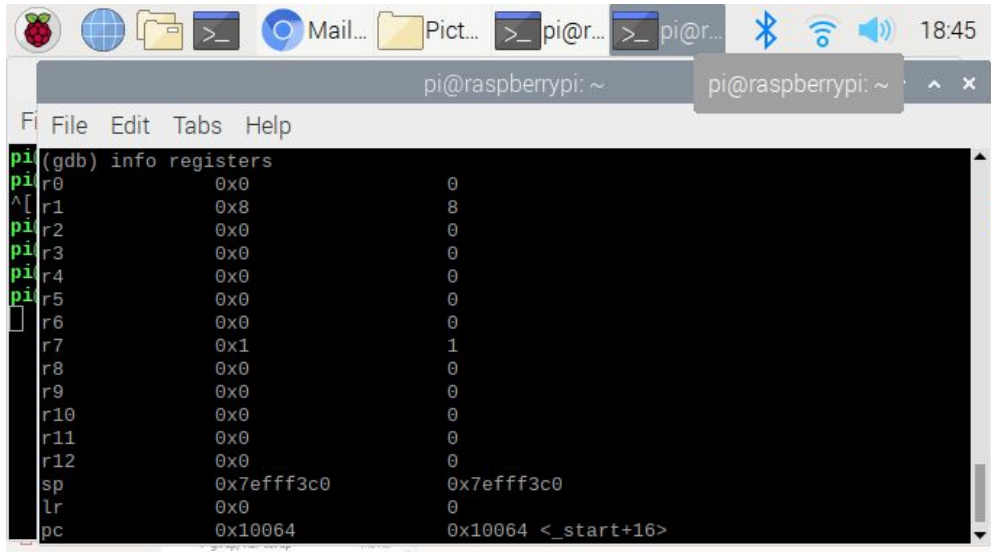
## First

```

pi@raspberrypi: ~
File Edit Tabs Help
pi (gdb) list
pi warning: Source file is more recent than executable.
pi ^
pi 1      @ first program
pi 2      .section .data
pi 3      .section .text
pi 4      .global _start
pi 5      _start:
pi 6          mov r1,#5 @load r1 with 5
pi 7          sub r1,r1,#1 @subtract 1 from r1
pi 8          add r1,r1,#4 @add 4 to r1
pi 9
pi 10         mov r7,#1
pi (gdb)
pi 11         svc #0
pi 12         .end
pi 13
pi (gdb) b 11

```

This is a screen shot of the first program done on the raspberry pi, where I am already in the debugger and have shown the contents of my code. It also displays that I have set a breakpoint at line 10 after my last operation.



```
pi (gdb) info registers
pi r0          0x0          0
pi r1          0x8          8
pi r2          0x0          0
pi r3          0x0          0
pi r4          0x0          0
pi r5          0x0          0
pi r6          0x0          0
pi r7          0x1          1
pi r8          0x0          0
pi r9          0x0          0
pi r10         0x0          0
pi r11         0x0          0
pi r12         0x0          0
pi sp          0x7efff3c0    0x7efff3c0
pi lr          0x0          0
pi pc          0x10064       0x10064 <_start+16>
```

This screenshot is to show the registers and their contents. The value shown in register 1 is 8 because that is the value from the code.

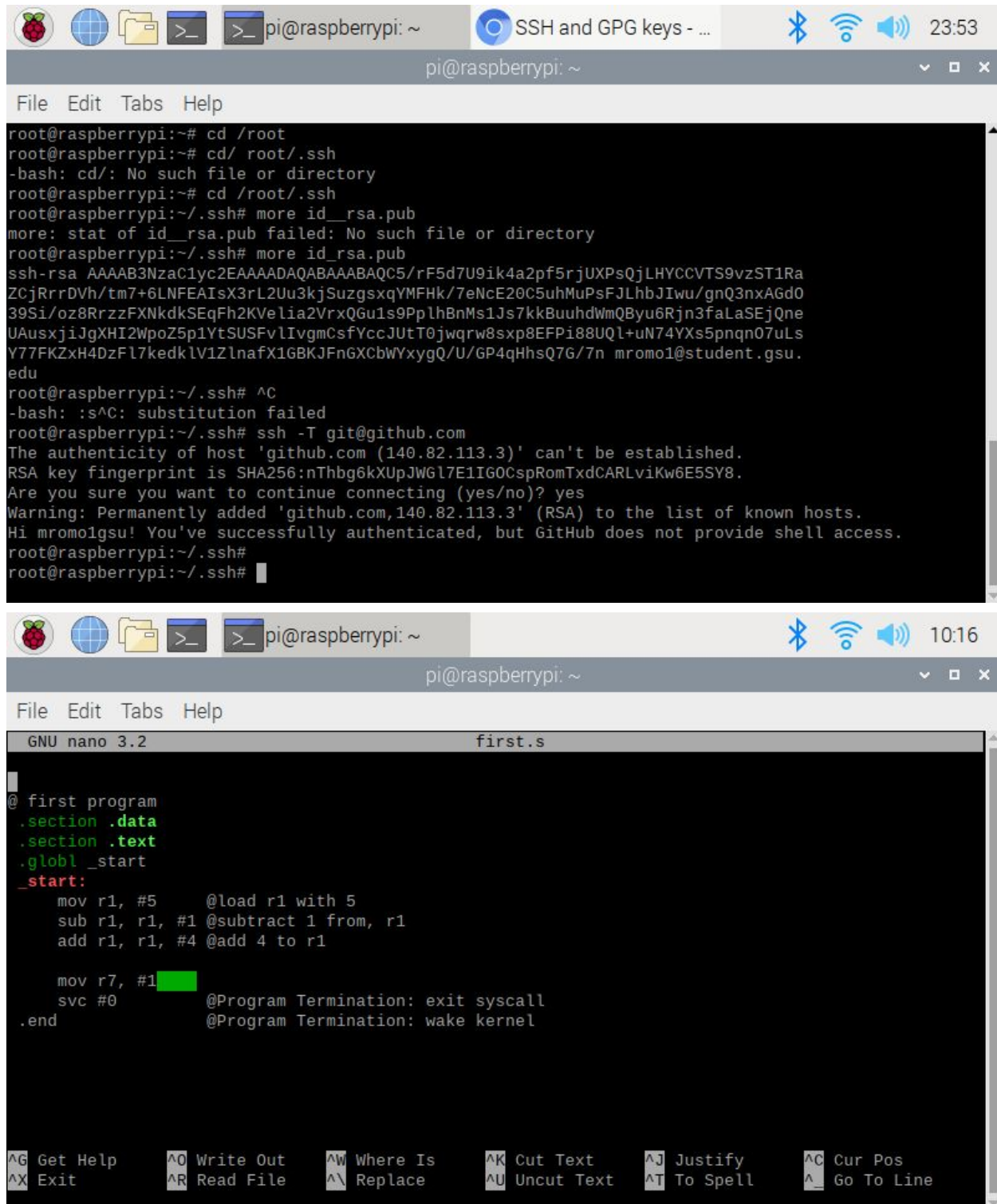


### **Teamwork basics: Miguel Romo**

- **What to do to get the task accomplished and the team members' satisfaction high?**
  - Two things get accomplished in good teams: the task gets accomplished and the satisfaction of team members is high.
- **Work Norms:**
  - **How will work be distributed?**
    - The work will be shared, with equal shares being distributed throughout the group.
  - **Who will set deadlines?**
    - The group will come to a decision on deadlines and make sure everyone is in agreement.
  - **What happens if someone doesn't follow through on his/her commitment (for example, misses a deadline)?**
    - We will discuss as a group why the deadline was not met and help the individual catch up. If they refuse to help then they will not receive credit for the work.
  - **How will the work be reviewed?**
    - There will be a second person reviewer followed by the group overview.
  - **What happens if people have different opinions about the quality of the work?**
    - The team will talk about the work and decide if the work needs to be redone.
  - **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).**
    - Different work habits are not an issue as long as they don't affect the overall group deadline. If the habits do affect the team, then as long as we are kept in the loop with the progress we can still talk about the issue and help the person complete the task earlier.
- **Facilitator Norms:**
  - **Will you use a facilitator?**
    - Yes, we will.
  - **How will the facilitator be chosen?**
    - Whomever volunteers for the position can be the facilitator.
  - **Will you rotate the position?**
    - The position will rotate for every project.
  - **What are the responsibilities of the facilitator?**
    - The responsibilities of the facilitator are Focus the team on the task get participation from all team members, keep the team to its agreed-upon time frame, suggest alternative procedures when the team is stalled, help team members confront problems, and summarize and clarify the team's decisions.
- **Communication Norms:**

- **When should communication takes place and through what medium (e.g., do some people prefer to communicate through e-mail while others would rather talk on the phone)?**
  - Groupme will be the source of the team's communication.
- **As a team, select two cases out of the four mentioned in Handling Difficult Behavior:**
  - Too quiet - the person doesn't like to share or does not want to share. A way to solve this is by making the member feel included and reassuring them that no idea is bad.
  - Argues - the person is being difficult when ideas are presented and alternative solutions are not offered. A way to solve this is to offer their own solutions when criticizing other ideas.
- When making decisions, If the team is having trouble reaching consensus, what should you do?
  - If this happens, we will determine if the decision needs to be voted on. If it is important, we will vote on a decision that is the best compromise.
- 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?
  - It will be the facilitators responsibility to ensure decisions are not made too quickly.
- 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 this happens, we can talk as a group and we can explain why they want that particular grade and try to communicate why they want the grade they want.

**Raspberry PI Installation and ARM Assembly Programming: Miguel Romo**



The image shows two screenshots of a Raspberry Pi terminal window. The top screenshot shows a user attempting to connect to a remote host via SSH. The user is in the root directory and tries to run 'cd /root/.ssh', which fails because the directory does not exist. They then try to view the contents of 'id\_rsa.pub' using 'more', which also fails. Finally, they attempt to connect to 'github.com' using 'ssh -T git@github.com'. The terminal shows the SSH warning about the host's authenticity and that GitHub does not provide shell access. The bottom screenshot shows the same terminal window with the GNU nano 3.2 editor open, editing a file named 'first.s'. The code in the file is assembly for a program that loads the value 5 into register r1, subtracts 1, adds 4, and then terminates the program using 'svc #0'.

```

root@raspberrypi:~# cd /root
root@raspberrypi:~# cd /root/.ssh
-bash: cd/: No such file or directory
root@raspberrypi:~# cd /root/.ssh
root@raspberrypi:~/.ssh# more id_rsa.pub
more: stat of id_rsa.pub failed: No such file or directory
root@raspberrypi:~/.ssh# more id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAC5/rF5d7U9ik4a2pf5rjUXPsQjLHYCCVTS9vzST1Ra
ZCjRrrDVh/tm7+6LNFEAIsX3rL2Uu3kjSuzgsxqYMFhk/7eNcE20C5uhMuPsFJLhbJIWu/gnQ3nxAGd0
39Si/oz8RrzzFXNkdKSEqFh2KVelia2VrxQGGu1s9PplhBnMs1Js7kkBuuhdWmQByu6Rjn3faLaSEjQne
UAusxjiJgXHI2WpoZ5p1YtSUSFvIvgmCsfYccJUtT0jwqrw8sxp8EFPi88UQl+uN74YXs5pnqn07uLs
Y77FKZxH4DzF17kedkLV1ZlnafX1GBKJFngXCbwYxygQ/U/GP4qHhsQ7G/7n mromo1@student.gsu.
edu
root@raspberrypi:~/.ssh# ^C
-bash: :s^C: substitution failed
root@raspberrypi:~/.ssh# ssh -T git@github.com
The authenticity of host 'github.com (140.82.113.3)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGl7E1IGOCspRomTxdCARLviKw6E5SY8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,140.82.113.3' (RSA) to the list of known hosts.
Hi mromo1gsu! You've successfully authenticated, but GitHub does not provide shell access.
root@raspberrypi:~/.ssh#
root@raspberrypi:~/.ssh#

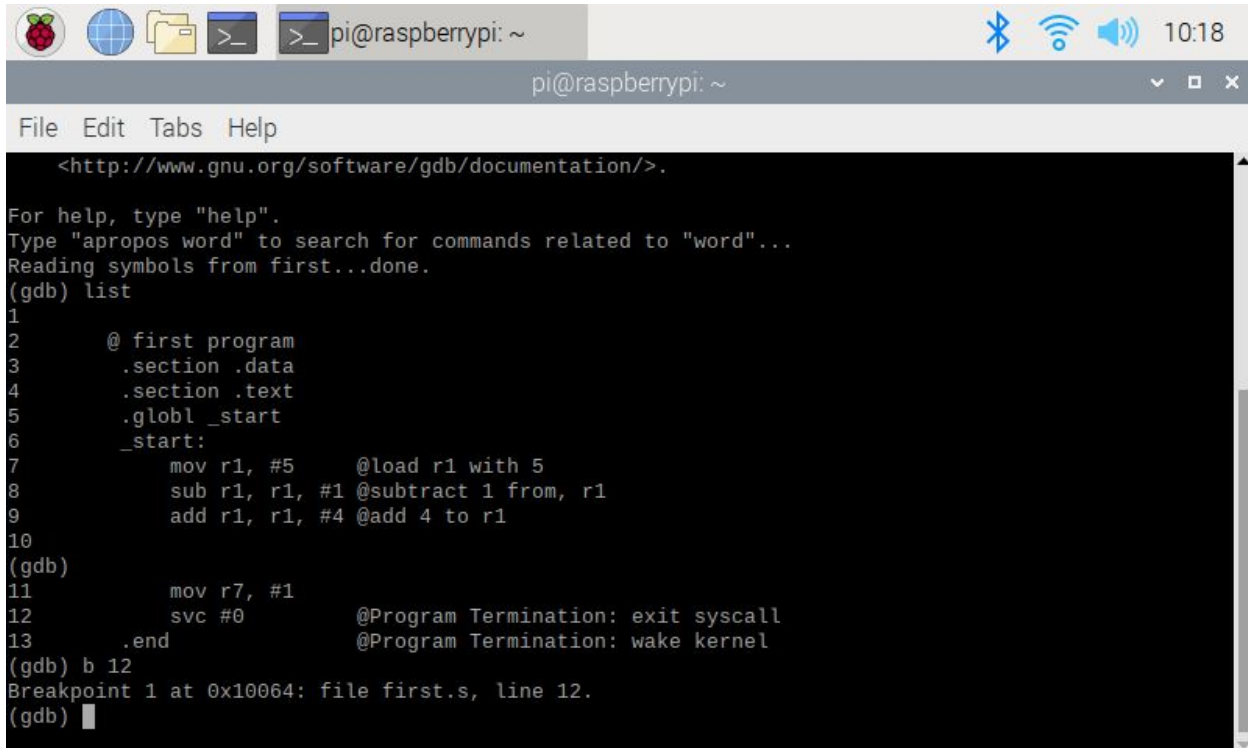
GNU nano 3.2                                first.s
@ first program
.section .data
.section .text
.globl _start
_start:
    mov r1, #5      @load r1 with 5
    sub r1, r1, #1  @subtract 1 from, r1
    add r1, r1, #4  @add 4 to r1

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

^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

```

Code for the First program

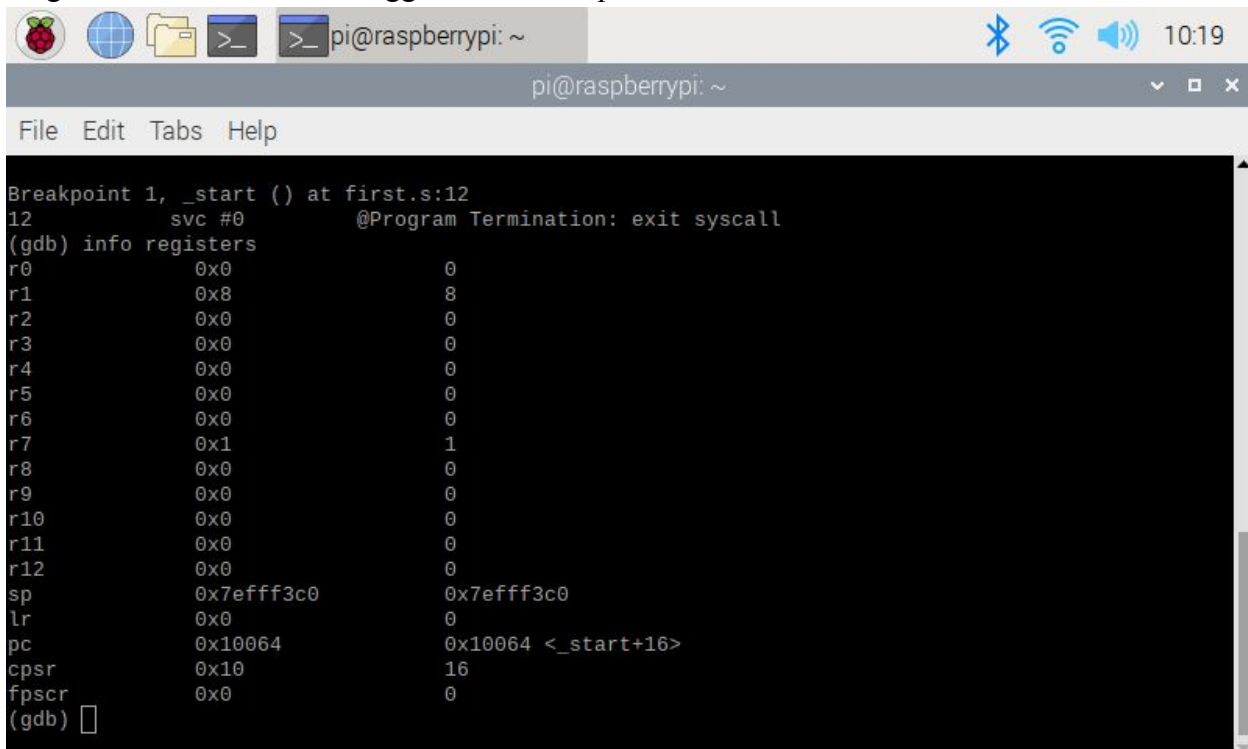


```

pi@raspberrypi: ~
File Edit Tabs Help
<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
2      @ first program
3      .section .data
4      .section .text
5      .globl _start
6      _start:
7          mov r1, #5      @load r1 with 5
8          sub r1, r1, #1 @subtract 1 from, r1
9          add r1, r1, #4 @add 4 to r1
10
11          mov r7, #1
12          svc #0          @Program Termination: exit syscall
13      .end                @Program Termination: wake kernel
(gdb) b 12
Breakpoint 1 at 0x10064: file first.s, line 12.
(gdb)

```

Program launched in the debugger and a breakpoint set to line 12.



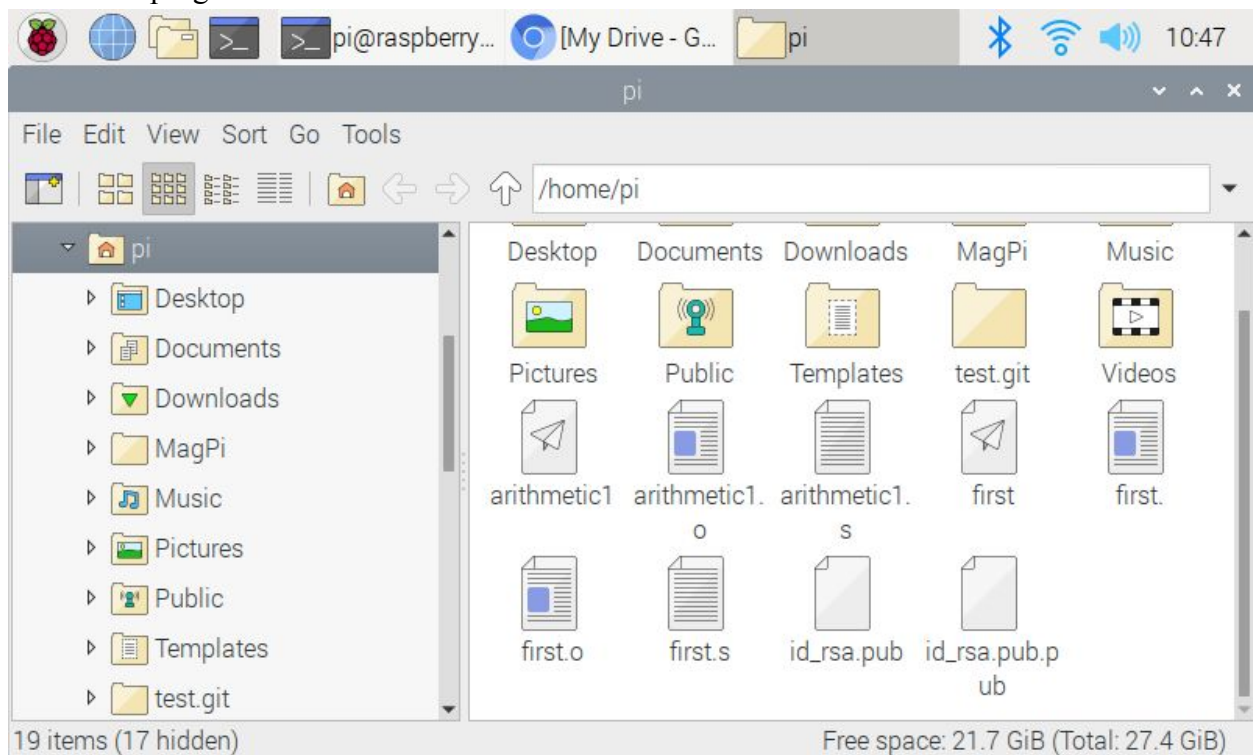
```

pi@raspberrypi: ~
File Edit Tabs Help
Breakpoint 1, _start () at first.s:12
12          svc #0          @Program Termination: exit syscall
(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)

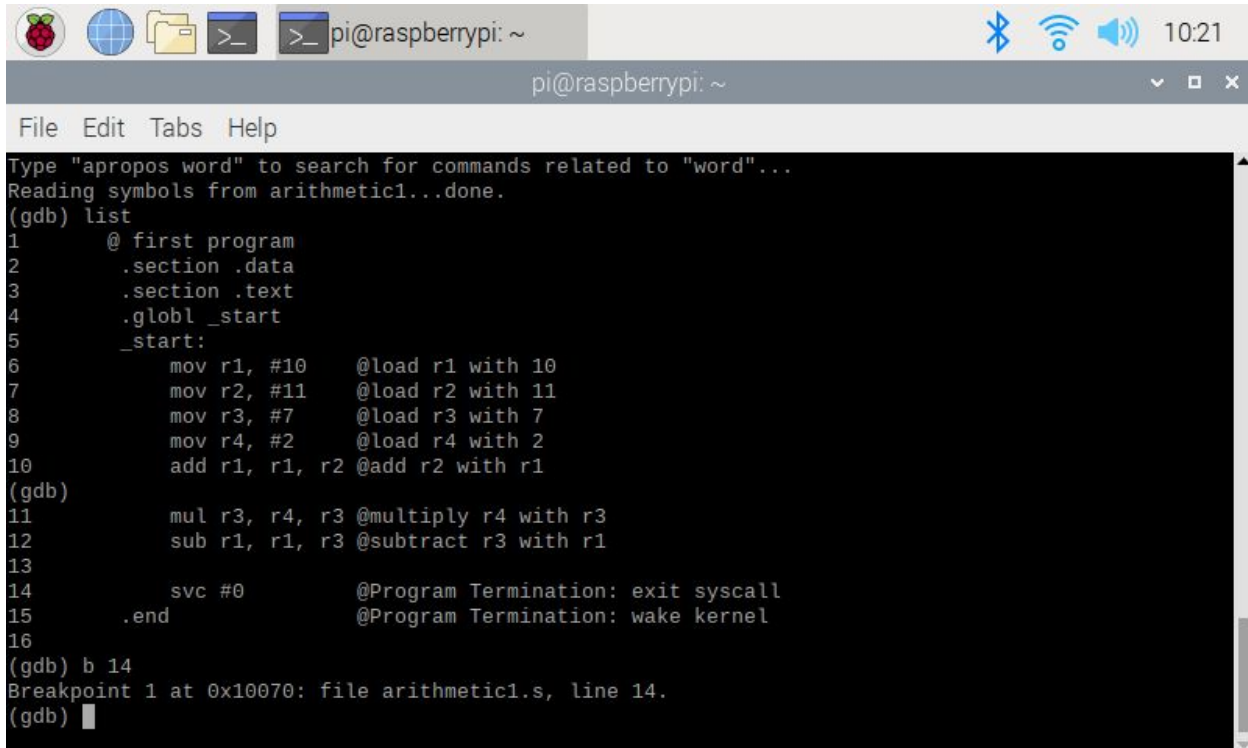
```

After we run the program we look at the registers to observe the contents. Here register r1 has a value of 8 because we subtracted 1 from the value in r1 and added 4. A result of 8 was stored in r1.

Here is the program for arithmetic.



The program was assembled and linked and the files were created.

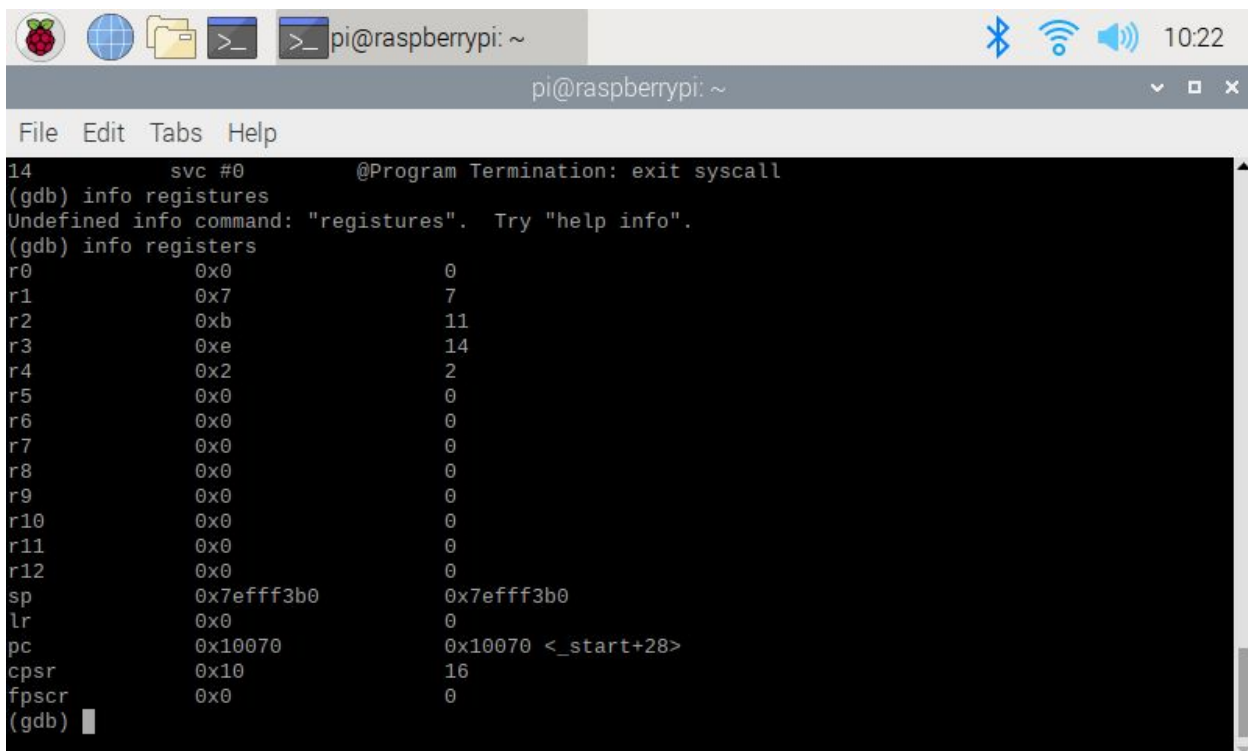


```

pi@raspberrypi: ~
File Edit Tabs Help
Type "apropos word" to search for commands related to "word"...
Reading symbols from arithmetic1...done.
(gdb) list
1      @ first program
2      .section .data
3      .section .text
4      .globl _start
5      _start:
6          mov r1, #10    @load r1 with 10
7          mov r2, #11    @load r2 with 11
8          mov r3, #7     @load r3 with 7
9          mov r4, #2     @load r4 with 2
10         add r1, r1, r2 @add r2 with r1
(gdb)
11         mul r3, r4, r3 @multiply r4 with r3
12         sub r1, r1, r3 @subtract r3 with r1
13
14         svc #0         @Program Termination: exit syscall
15     .end               @Program Termination: wake kernel
16
(gdb) b 14
Breakpoint 1 at 0x10070: file arithmetic1.s, line 14.
(gdb)

```

The debugger is opened to observe the memory, and study the registers. A break point is added at line 14.



```

pi@raspberrypi: ~
File Edit Tabs Help
14         svc #0         @Program Termination: exit syscall
(gdb) info registers
Undefined info command: "registers". Try "help info".
(gdb) info registers
r0          0x0          0
r1          0x7          7
r2          0xb          11
r3          0xe          14
r4          0x2          2
r5          0x0          0
r6          0x0          0
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)

```

In the info register window we see what values are registers hold. R1 had the value of 7 because we added 11 to the initial value in register r1, which was 10. Finally we subtract the value of r3

from r1, 21 - 14, which is 7. R2 & r4 holds the initial values we stored in it using “mov”, r3 had the value of 7, but now hold 14, since we multiplied  $7 * 14$ .

## Appendix:

Slack: [the-commuters.slack.com](https://the-commuters.slack.com)

<https://app.slack.com/client/TSWLWS9LK/DTAC2TKMM>

The screenshot shows a Slack channel conversation. At the top, it says "Thursday, January 30th". The channel name is "#team-intro".

- Joan Galicia** 10:09 PM: joined #team-intro.
- Joan Galicia** 10:15 PM: Hello, My name is Joan Galicia and I am a computer science major and am interested in computer programming. My task is to create a slack account and to have my team members write an introduction about themselves and what their task is in the group assignment. I expect to gain more communication with my peers and to understand how this team will function as we work on our projects together.
- Team Members will write an introduction which is their name, interest, assigned task/s, expectation, from this project.
- Andre Nguyenphuc** 10:44 PM: joined #team-intro along with 2 others.

---

**Thursday, January 30th**

- Arteen Ghafourikia** 10:41 AM: Hello, My name is Arteen Ghafourikia and I am a computer science major, and I am interested in making games. My task is to connect the Raspberry pi to GitHub, add members to it, and write the assembly code. I expect to learn how to problem solve and work as a group more proficiently. (edited)
- Alaya Shack** 1:51 PM: Hello, my name is Alaya Shack, and I am interested in data science and project management. My task is to format and write out our answers to the Teamwork Basics Document. I expect to enhance my communication skills with my peers, gain more experience with cooperative learning, and to further improve my problem solving skills with ARM assembly programming.
- Andre Nguyenphuc** 8:12 PM: Hello, My name is Andre Nguyenphuc. I am a computer science major and I am interested in data science. My task is to write up the report for the group. I expect to learn how a team is supposed to work together to achieve a common goal.

---

**Yesterday**

- Miguel** 4:17 PM: joined #team-intro.
- Miguel** 4:18 PM: Hello, Everyone. My my name is Miguel Romo and I am a computer science major, and I'm interested in cybersecurity. My task is planning and scheduling. I will be creating a table with everyone's name assigned tasks. I expect to learn how projects are accomplished using teamwork.

Github: <https://github.com/Arteenghafourikia/CSC3210-TheCommuters>

The screenshot shows a GitHub repository page for "Arteenghafourikia / CSC3210-TheCommuters".

- Repository name: **CSC3210-TheCommuters**
- Branch: **master**
- File: **CSC3210-TheCommuters / README.md**
- Commit: **andrenghuyenphuc Update README.md** (7eab10b now)
- Contributors: 2 contributors
- File size: 3 lines (3 sloc) | 229 Bytes
- Actions: Raw, Blame, History

**CSC3210-TheCommuters**

GSU CSC3210 2020\_SPRING Group Project Developing Soft and Parallel Programming Skills Using Project-Based Learning

Team Members: Arteen Ghafourikia, Alaya Shack, Andre Nguyenphuc, Joan Galicia, Miguel Romo

© 2020 GitHub, Inc. | Terms | Privacy | Security | Status | Help | Contact GitHub | Pricing | API | Training | Blog | About



Arteenghafourikia / CSC3210-TheCommuters

<> Code ① Issues 0 📄 Pull requests 0 ⚙️ Actions 📁 Projects 1 📖 Wiki 🛡️ Security 📊 Insights ⚙️ Settings

Project\_1A  
Updated 13 hours ago

2 To Do + ...

📝 Edit the Video ...

Added by Arteenghafourikia

📝 Film the Presentation ...

Added by Arteenghafourikia

3 Progress + ...

📝 Finish the Report ...

Added by Arteenghafourikia

📝 Update Planning and Schedule ...

Added by mromo1gsu

📝 Put the Everything together ...

Added by Arteenghafourikia

3 Done + ...

📝 Upload video on youtube ...

Added by Arteenghafourikia

📝 Setup Youtube ...

Added by Arteenghafourikia

📝 Github and Everyone added. Assembly code snippets and observations. ...

Added by Arteenghafourikia

## Youtube Video

<https://www.youtube.com/watch?v=0b5vlos9Q54>