

Developing Soft and Parallel Programming Skills Using Project-Based Learning

Justin Choi, Adam Henrie, Jean-Pierre Sacha, Titilayo Shonuyi, Shanza Siddiqi

Spring 2019
Group Name: Qubits

Planning and Scheduling

Team name: qubits

Name	Email	Task	Duration	Dependency	Due Date	Note
Justin Choi	jchoi34@student.gsu.edu	Report	3 hours	Task 3 and Task 4	2/21/19	Report must be finished the day before the due date
Adam Henrie	ahenrie1@student.gsu.edu	Uploading video, Parallel programming	4 hours	Video	2/21/19	Please send the link to the video presentation task 3 information
JP Sacha	jsacha2@student.gsu.edu	Help with arm assembly programming and report	4 hours	None	2/21/19	Please assist in assembly programming
Titilayo Shonuyi	tshonuyi1@student.gsu.edu	Arm assembly programming	3 hours	None	2/21/19	Please send task 4 information for the report
Shanza Siddiqi	Ssiddiqi2@student.gsu.edu	Create To do/In Progress/Done columns and cards	3 hours	None	2/18/19	Please create new cards on github and provide screenshots of the tasks

Teamwork Basics

1. What to do to get the task accomplished *and* the team members' satisfaction high?

To keep the team member satisfaction high we can, set ground rules, use a facilitator, know the strengths of your group members, keep lines of communication open, and know how to avoid or solve common problems. We have set ground rules for tangents and overly talkative people. We have also acknowledged the need for a group facilitator. The position is on a volunteer basis.

2. Answer all the questions in the Work Norms, Facilitator Norms, Communication Norms using your own words and your own context.

- **Work Norms:** How will work be distributed? Who will set deadlines? What happens if someone doesn't follow through on his/her commitment (for example, misses a deadline)? How will the work be reviewed? What happens if people have different opinions about the quality of 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).

The group will volunteer for each task and will meet together to decide on the deadlines. If someone doesn't follow through on their commitment, the team will try to talk to that person about any problems they are facing and if they need any help with their task. All files can be shared through GitHub, so everyone is able to view the work that is being done and give input on the work. We all have similar work habits and will work consistently to finish the project. i.e. We have agreed to meet up at predetermined times and are aware that some work may be telecommuted from within the Slack platform.

- **Facilitator Norms:** Will you use a facilitator? How will the facilitator be chosen? Will you rotate the position? What are the responsibilities of the facilitator? (see below)

JP has basically taken on the facilitator role right from the start and has kept the team progressing. The role of the facilitator is to focus the team on the task, get participation from all team members, keep the team to adhere to their deadlines, suggest alternative procedures when the team isn't progressing, help team members confront problems, and summarize and clarify the team's decisions. Facilitator will be a voluntary role.

- **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)?

Communication should take place regularly in order to be up to date on everyone's progress. It should be done through Slack since it is the easiest way to contact every team member since we are all using it.

- **Meeting Norms:** What is everyone's schedule? Should one person be responsible for coordinating meetings? Do people have a preference for when meetings are held? Where is a good place to hold meetings? What happens if people are late to a meeting? What happens if a group member misses a meeting? What if he/ she misses several meetings?

Since everyone has a different schedule, it is a collaborative effort to coordinate meetings. A good place to hold meetings is in the library. If a group member misses a meeting, it is his or her duty to inform the group and to make sure he or she gets all the information from the meeting. If a group member makes it a regular occurrence to miss meetings, then the rest of the group will confront the individual to ascertain what is going on.

- **Consideration Norms:** Can people eat at meetings? smoke? What happens if someone is dominating the discussion? How can norms be changed if someone is not comfortable with what is going on in the team?

It is okay to eat in meetings, but it is definitely not okay to smoke. If someone is dominating a discussion, the facilitator could step in and allow other members to provide their input. If any group member is uncomfortable with anything that is going on in the team, then the group can alter current protocols to accommodate that individual.

3. As a team, select two cases out of the four mentioned in Handling Difficult Behavior. (use your own words and your own context)

Too quiet: The facilitator or any other group member could ask for the quiet individual's input during discussions if they are being noticeably quiet.

Overly talkative: If a certain member is being too talkative, someone can ask for additional input from the rest of the group to try and give someone else the spotlight.

4. When making decisions, if the team is having trouble reaching a consensus, what should you do? (use your own words and your own context)

If the group is having difficulties in reaching a consensus when making decisions, the group can compromise by voting on the most popular option. Any other dissention from within the group will be discussed from within Slack if a group member is still not cooperative.

5. What should you 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?

If a person is pressuring people to move on before it's a good idea to do so, the group can agree to come back to the current decision at a later time in order to discuss it more. We will use Slack for this purpose. Which will allow us the opportunity to move forward with other tasks but still be able to address everyone's concerns eventually.

6. 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 the majority of the group wants to obtain an “A”, but there is someone that is satisfied with only a “B”, then the group can assign that individual somewhere between 0-100% on the note column of the report, based on their level of cooperation and contribution to the team’s workload. Group consensus is that unless situations arise that prevent the entire team from getting a high grade, the grade goal is within the range of B to A.

Parallel Programming Skills

Foundation

1. Identifying the components on the raspberry PI B+

Raspberry Pi B+ has 1gb of RAM, a quad core ARM based processor at 1.4ghz, a micro-usb power supply, 3.5mm audio out jack, HDMI out jack, Gigabit Ethernet, and 4 USB 2.0 ports, there is also a GPIO header and camera header on the motherboard.

2. How many cores does the Raspberry Pi's B+ CPU have?

The B+ has a quadcore cpu at 1.4ghz clock speed.

3. List three main differences between X86 (CISC) and ARM Raspberry PI (RISC).

Justify you answer and use your own words (do not copy and past)

X86 (CISC) is considered to have a register memory structure whereas ARM(RISC) is a register-register implementation. CISC is considered more flexible because it allows for a more complex instruction set, however it has less registers than ARM. ARM uses Big endien notation but allows for switching between endiens. Finally ARM is considered faster in some respects because of less instructions needed to perform a task. However creative programming needs to be implemented in order to get similar overall results to that of x86.

4. What is the difference between sequential and parallel computation and identify the practical significance of each?

Sequential computation is good for code that cannot be easily broken up into parallel chunks. Parallel is good for chunks of code that can be parallelized such as loops.

5. Identify the basic form of data and task parallelism in computational problems.

Data parallelism is defined as parallel calculations that can performed to compute data in parallel. Task parallelism involves not just parallel data computations but the overall set of tasks that occur can be parallelized i.e. a whole set of equations. The main distinction here is that data parallelism happens to a strict set of data whereas Task is more broad.

6. Explain the differences between processes and threads.

Processes are distinct instances of a program that perform a myriad of tasks, there can be sub processes in a process that allow for discrete processing however they should not be confused with threads which allow for multiple instances of code for a process to be split and be processed in parallel on multicore cpus before being joined after they output the desired result.

7. What is OpenMP and what is OpenMP pragmas?

OpenMP is the framework designed for the C language that allows for implicit multi-threading and parallelism in processes. It is implicit because it allows for a pre-built framework to code and utilize multithreading through the OS itself. This function is achieved via a compiler directive called “pragmas” that signals the OS to treat the code segment as parallel. It is achieved via a single task multiple data implementation.

8. What applications benefit from multi-core (list four)?

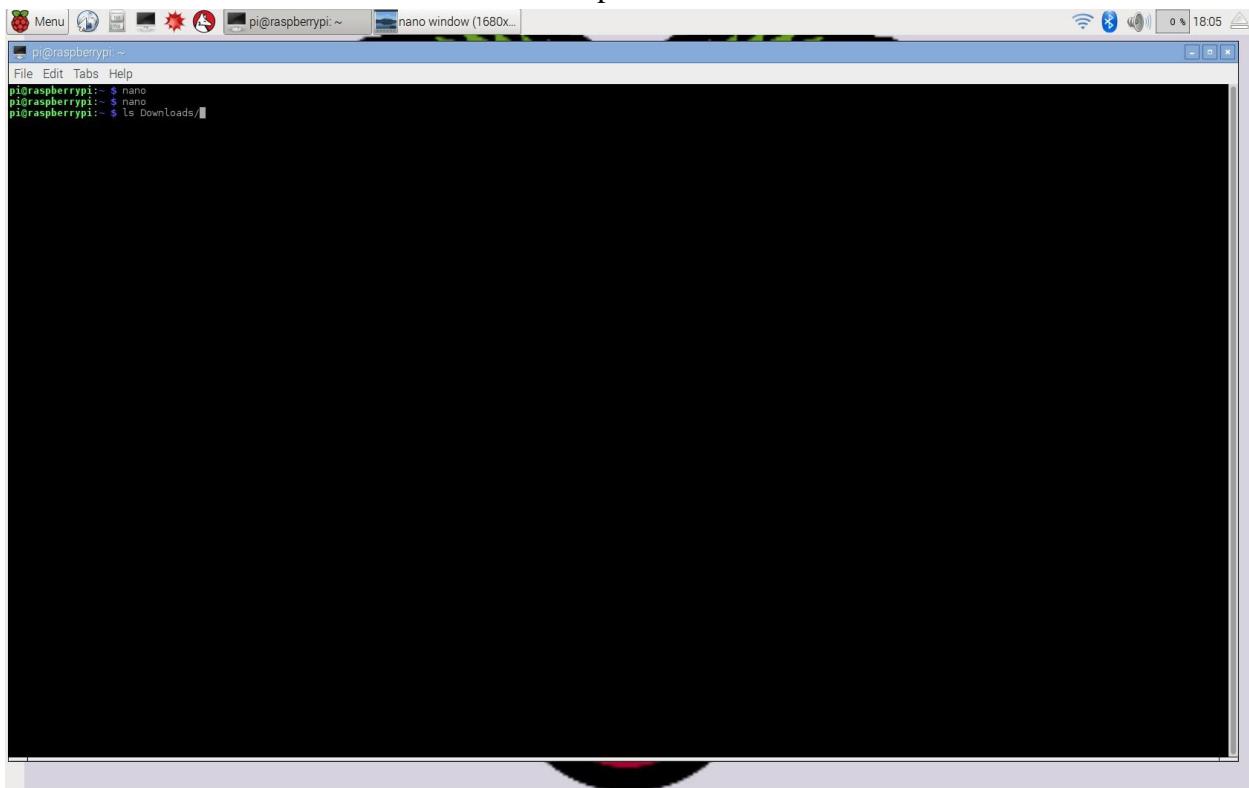
Applications that benefit heavily from multicore systems are Web browsers with multi-media plugins, web servers (where every new connection through a socket could be handled by a core on a multithreaded CPU), database servers which are also constantly working on multiple tasks at one time, and finally modern PC and console and mobile games that run on X86 and ARM architectures. Games such as Battlefield 3, and PUBG are designed from the ground up to utilize multi-core cpus with 4 and 6 cores respectively to process the games real time game engines.

9. Why Multicore? (why not single core, list four)

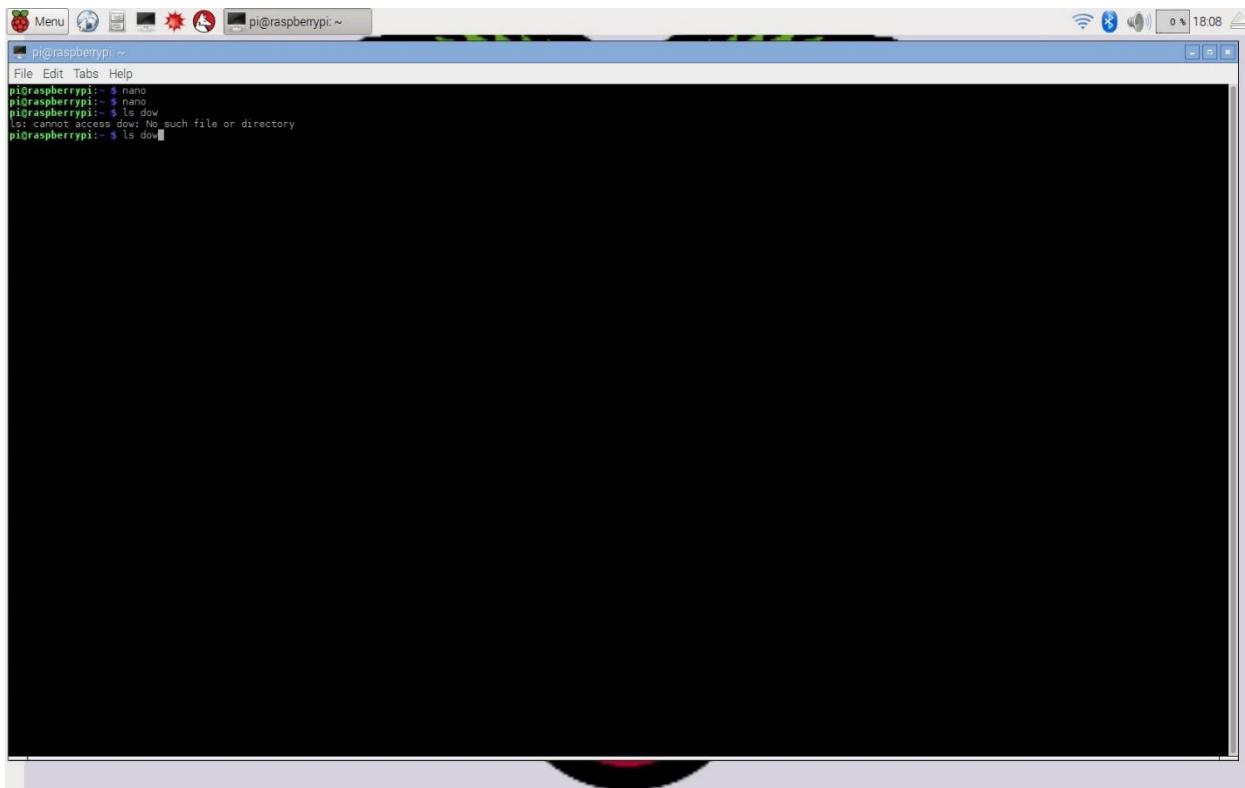
Multicore helps to solve the problem of increasingly difficult to achieve processor frequencies by splitting the workload. Decreases the need for dedicated server based computation by allowing for more speed on a personal workstation. Many applications that need more speed than can be supplied by a serialized single core processor benefit from multi-threading on multi-core CPUs. Multi-core also allows for the use of many separate serial computation only programs to happen on the same CPU die.

Parallel Programming Basics

Here we can see that we can tab and auto-complete a command such as ls Downloads



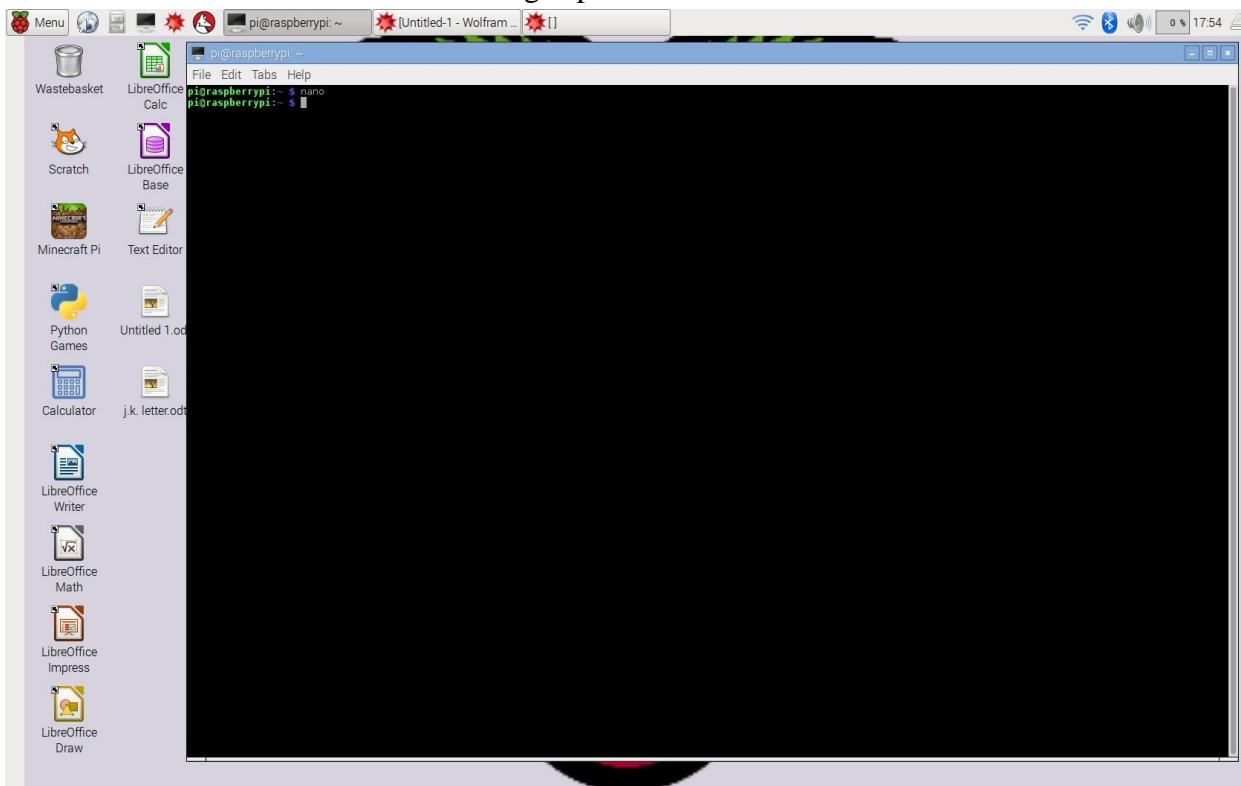
Here we can see that we can try a command and then hit the up arrow key in order to re-list the last command.



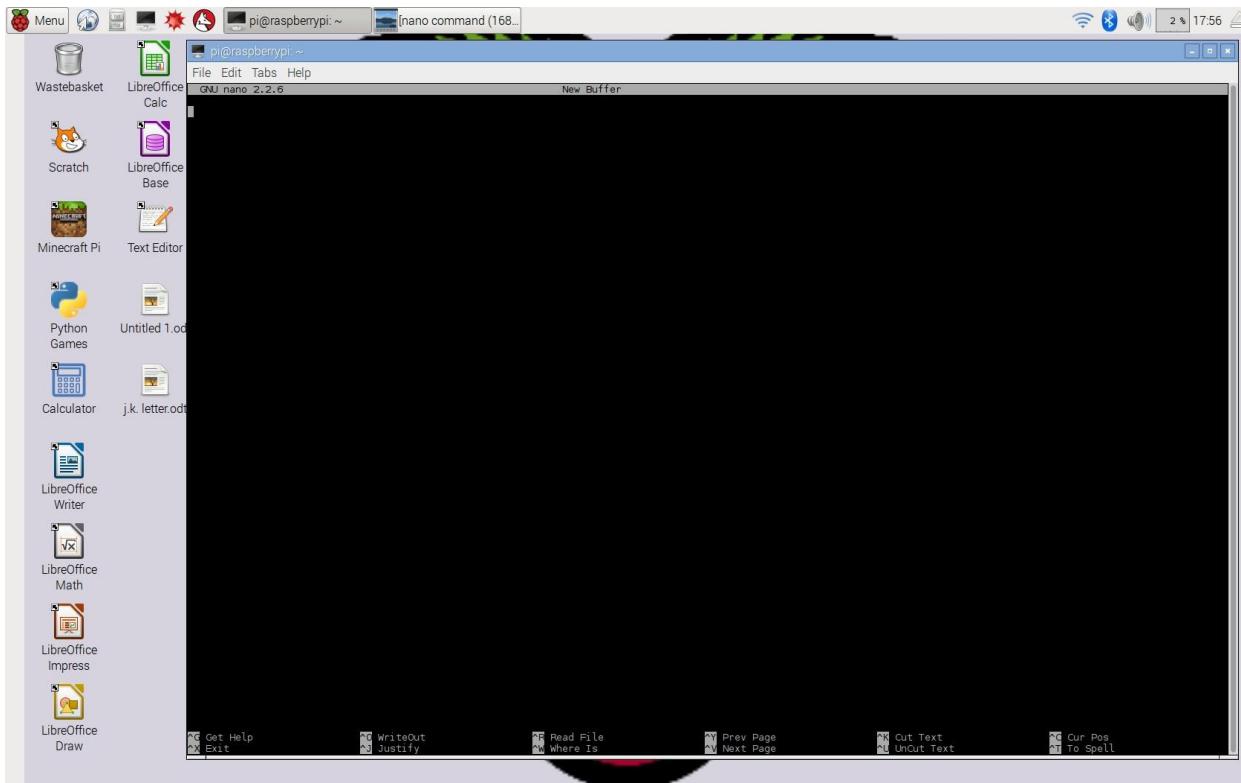
A screenshot of a terminal window titled "pi@raspberrypi: ~". The window shows a command-line interface with the following history:

```
pi@raspberrypi:~$ nano  
pi@raspberrypi:~$ nano  
pi@raspberrypi:~$ ls dow  
ls: cannot access dow: No such file or directory  
pi@raspberrypi:~$ ls dow
```

Here we can see the nano command brings up the nano editor



Here we can see the nano window



Here I input the command nano spmd2.c to create the file and then write this code.

```

pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.2.6
New Buffer Modified
#include <stdio.h>
#include <omp.h>
#include <stdlib.h>

int main(int argc, char** argv){
    int id, numThreads;
    printf("Hello\n");
    if (argc > 1){
        omp_set_num_threads(atoi(argv[1]));
    }
    #pragma omp parallel
    {
        id = omp_get_thread_num();
        numThreads = omp_get_num_threads();
        printf("Hello from thread %d of %d\n", id, numThreads);
    }
    printf("\n");
    return 0;
}

```

Next the code was compiled using this command: gcc spmd2.c -o spmd2 –fopenmpA compile error was encountered and then fixed.

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi: $ nano
pi@raspberrypi: $ nano
pi@raspberrypi: $ ls dow
ls: cannot access dow: No such file or directory
pi@raspberrypi: $ gpicview kf
pi@raspberrypi: $ gpicview kf
pi@raspberrypi: $ nano
pi@raspberrypi: $ nano spmd2.c
pi@raspberrypi: $ ls
scrat.png 2019-02-15-175634_1680x1050_scrat.png 2019-02-15-180843_1680x1050_scrat.png Desktop Downloads nukit Public Scratch Templates
2019-02-15-175419_1680x1050_scrat.png 2019-02-15-180658_1680x1050_scrat.png 2019-02-15-180956_1680x1050_scrat.png Documents Music Pictures python_games spmd2.c Videos
pi@raspberrypi: $ vi spmd2.c
[1]+ Stopped                  vi spmd2.c
pi@raspberrypi:~ $ vi spmd2.c
[2]+ Stopped                  vi spmd2.c
pi@raspberrypi: $ nano
pi@raspberrypi: $ nano
pi@raspberrypi: $ gcc spmd2.c -o spmd2 -fopenmp
spmd2.c: In function 'main':
spmd2.c:9:1: error: expected ';' before 'if'
if (argc > 1){
^
pi@raspberrypi: $ nano
pi@raspberrypi: $ gcc spmd2.c -o spmd2 -fopenmp
pi@raspberrypi:~ $ 

```

The program was run with an input of 4, 3, 2, 1. A critical section error occurred and the memory locations for each thread are shared and not mutually exclusive. Thus, the cores reported multiple duplicates of the same thread id.

```
Raspberry Pi Menu | Home | Help | pi@raspberrypi: ~

File Edit Tabs Help

pi@raspberrypi:~$ nano
pi@raspberrypi:~$ nano
pi@raspberrypi:~$ ls dow
ls: cannot access dow: No such file or directory
pi@raspberrypi:~$ gpicview %f
pi@raspberrypi:~$ gpicview %f
pi@raspberrypi:~$ nano
pi@raspberrypi:~$ nano spmd2.c
pi@raspberrypi:~$ gcc spmd2.c -o spmd2 -fopenmp
pi@raspberrypi:~$ ./spmd2 2
2019-02-15-175619_1680x1050_scren.png 2019-02-15-175634_1680x1050_scren.png 2019-02-15-180843_1680x1050_scren.png Desktop Downloads nukit Public Scratch Templates
2019-02-15-175619_1680x1050_scren.png 2019-02-15-180843_1680x1050_scren.png 2019-02-15-180856_1680x1050_scren.png Documents Music Pictures python_games spmd2.c Videos
pi@raspberrypi:~$ vi spmd2.c

[1]+ Stopped                 vi spmd2.c
pi@raspberrypi:~$ vi spmd2.c

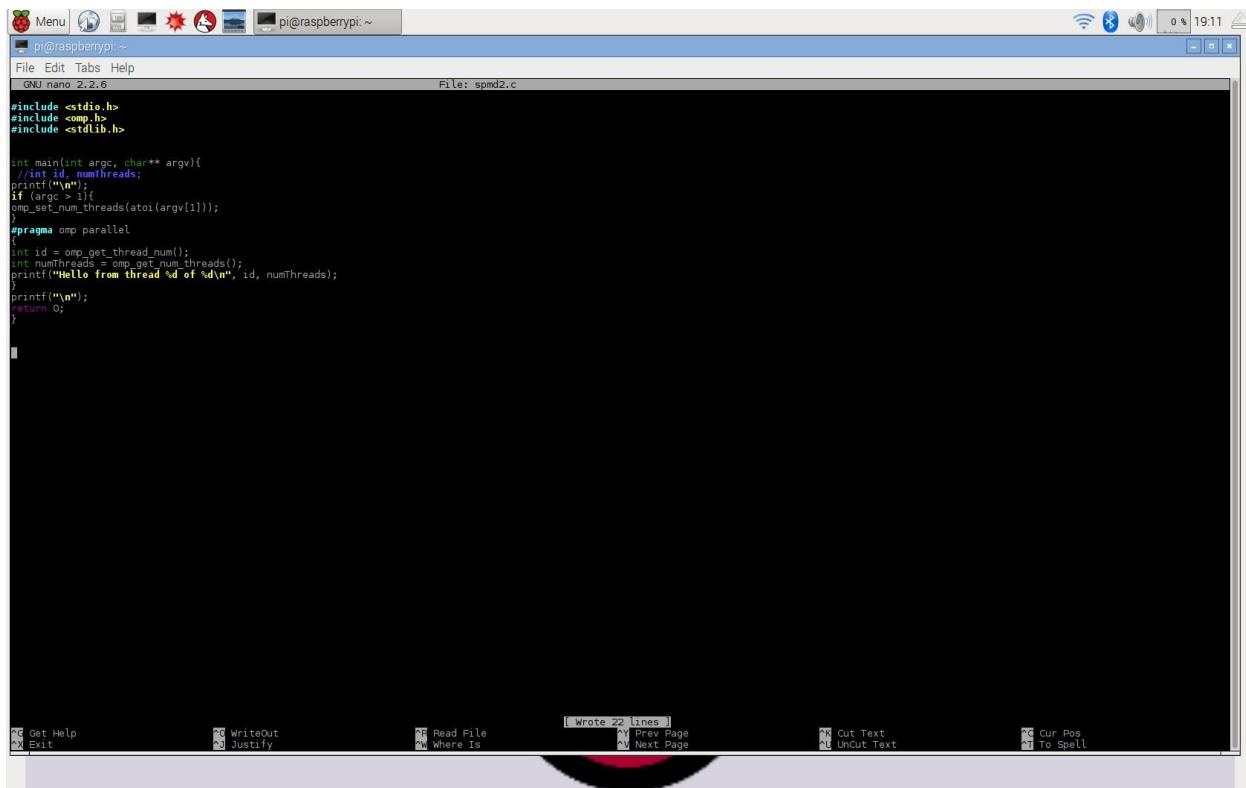
[2]+ Stopped                 vi spmd2.c
pi@raspberrypi:~$ nano
pi@raspberrypi:~$ nano
pi@raspberrypi:~$ gcc spmd2.c -o spmd2 -fopenmp
pi@raspberrypi:~$ ./spmd2 4
Hello from thread 2 of 4
Hello from thread 1 of 4
Hello from thread 0 of 4
Hello from thread 3 of 4

pi@raspberrypi:~$ ./spmd2 4

Hello from thread 2 of 4
Hello from thread 2 of 4
Hello from thread 2 of 4
Hello from thread 3 of 4

pi@raspberrypi:~$ ./spmd2 1
Hello from thread 0 of 1
pi@raspberrypi:~$ ./spmd2 2
Hello from thread 0 of 2
Hello from thread 1 of 2
pi@raspberrypi:~$ ./spmd2 3
Hello from thread 0 of 3
Hello from thread 0 of 3
Hello from thread 1 of 3
pi@raspberrypi:~$
```

Here, two lines of code were added to allow for separate memory location for each core. Two lines were fully declared with int. And the original int variable declaration of id and numThreads were commented out.



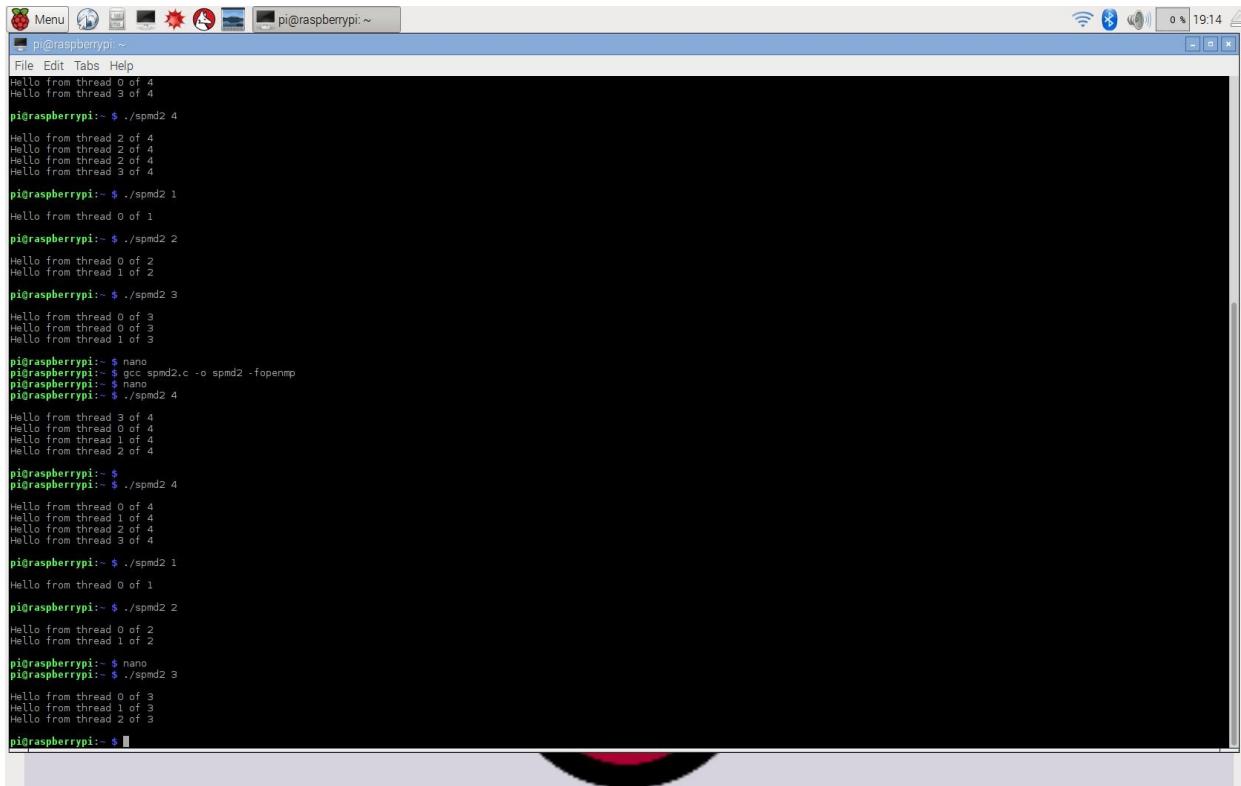
The screenshot shows a terminal window titled "pi@raspberrypi: ~" running the "nano 2.2.6" text editor. The file being edited is "spmd2.c". The code contains the following content:

```
#include <stdio.h>
#include <omp.h>
#include <stdlib.h>

int main(int argc, char** argv){
    //int id, numThreads;
    printf("Hello from thread %d of %d\n", id, numThreads);
    if(argc > 1){
        omp_set_num_threads(atoi(argv[1]));
    }
    #pragma omp parallel
    {
        int id = omp_get_thread_num();
        int numThreads = omp_get_num_threads();
        printf("Hello from thread %d of %d\n", id, numThreads);
    }
    printf("\n");
    return 0;
}
```

The terminal window includes standard Linux system icons at the top and a toolbar at the bottom with various file operations like "Get Help", "Exit", "WrittenOut", "Justify", "Read File", "Where Is", "Wrote 22 lines", "Prev Page", "Next Page", "Cut Text", "Uncut Text", "Cur Pos", and "To Spell".

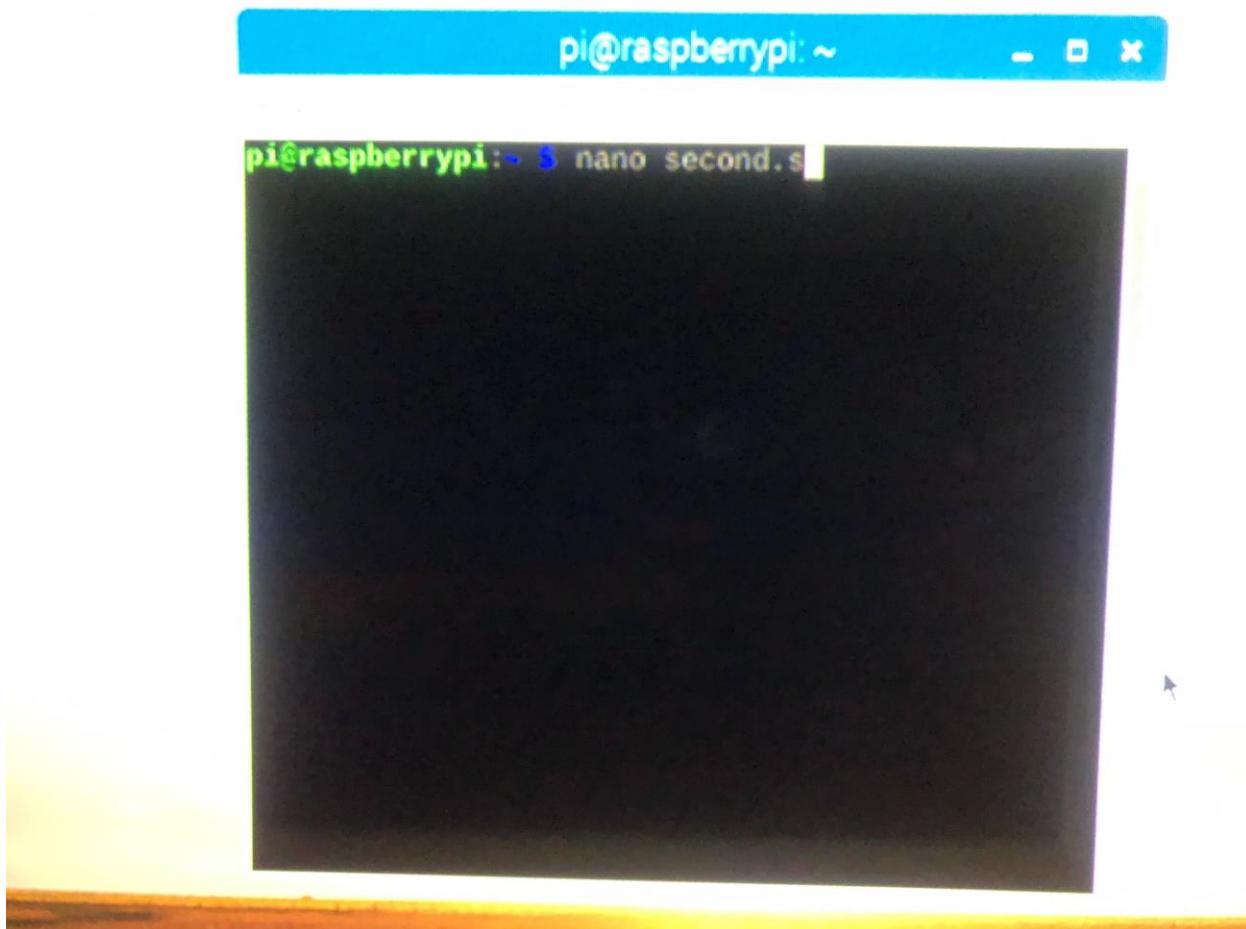
Here, below. The program has been recompiled and we have re-input 1-4 and each core is reporting in with its own unique thread id out of four.



The screenshot shows a terminal window titled "pi@raspberrypi: ~" running on a Raspberry Pi. The window displays a series of "Hello from thread X of Y" messages, where X and Y range from 0 to 4. This indicates that the program is utilizing four cores (Y=4) and has four threads per core (X=0 to 3). The terminal also shows the compilation command "gcc spmd2.c -o spmd2 -fopenmp" and the execution of the program with ./spmd2 followed by arguments 1, 2, 3, and 4. The window has a standard Linux-style header bar with icons for menu, file, edit, tabs, help, and system status.

```
pi@raspberrypi: ~
File Edit Tabs Help
Hello from thread 0 of 4
Hello from thread 3 of 4
pi@raspberrypi: $ ./spmd2 4
Hello from thread 2 of 4
Hello from thread 2 of 4
Hello from thread 2 of 4
Hello from thread 3 of 4
pi@raspberrypi: $ ./spmd2 1
Hello from thread 0 of 1
pi@raspberrypi: $ ./spmd2 2
Hello from thread 0 of 2
Hello from thread 1 of 2
pi@raspberrypi: $ ./spmd2 3
Hello from thread 0 of 3
Hello from thread 0 of 3
Hello from thread 1 of 3
pi@raspberrypi: $ nano
pi@raspberrypi: $ gcc spmd2.c -o spmd2 -fopenmp
pi@raspberrypi: $ ./spmd2 4
Hello from thread 3 of 4
Hello from thread 0 of 4
Hello from thread 1 of 4
Hello from thread 2 of 4
pi@raspberrypi: $ ./spmd2 1
Hello from thread 0 of 1
pi@raspberrypi: $ ./spmd2 2
Hello from thread 0 of 2
Hello from thread 1 of 2
pi@raspberrypi: $ nano
pi@raspberrypi: $ ./spmd2 3
Hello from thread 0 of 3
Hello from thread 1 of 3
Hello from thread 2 of 3
pi@raspberrypi: $
```

ARM Assembly Programming



Writing the program.

The screenshot shows a terminal window titled "pi@raspberrypi: ~". Inside the window, the nano text editor is open with the file "second." modified. The assembly code in the editor is:

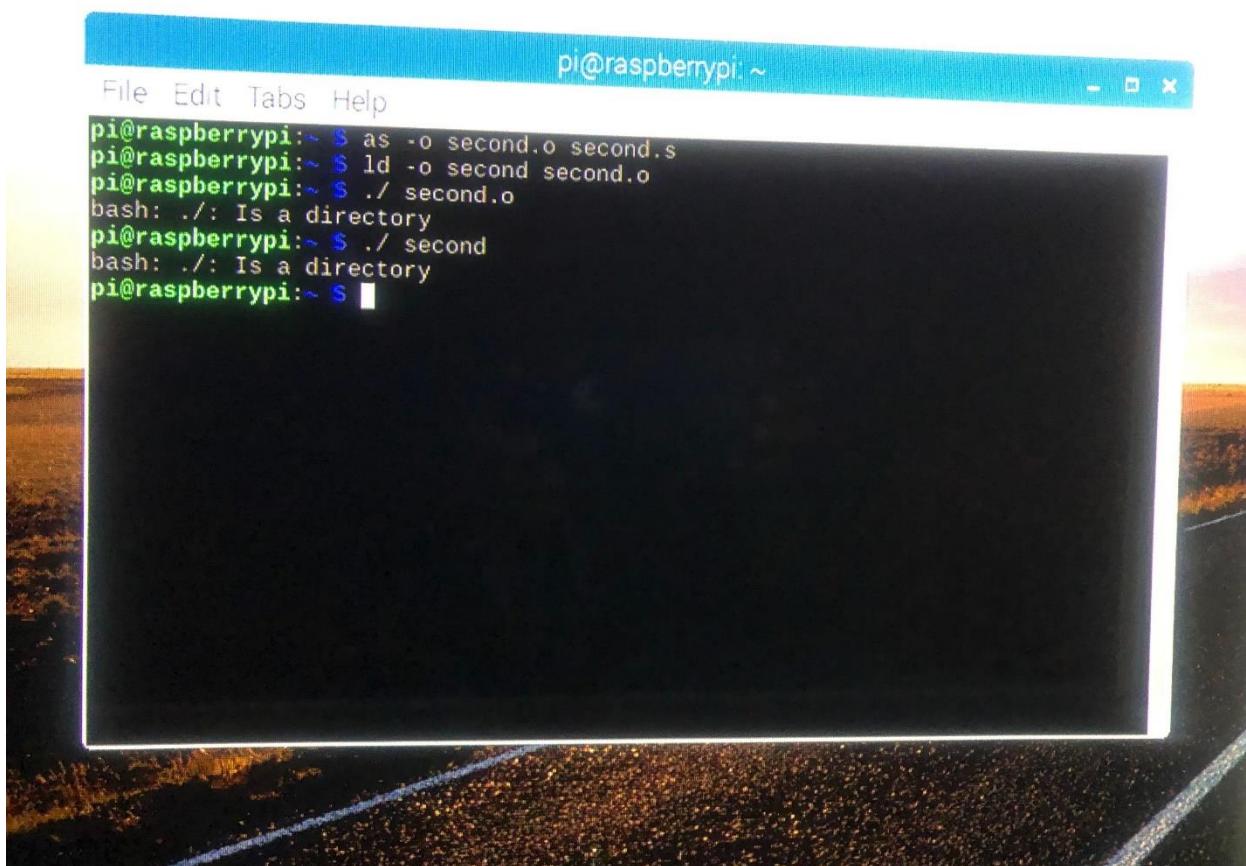
```
second program: c= a+b
.section .data
a: .word 2
b: .word 5
c: .word 0
.section .text
.globl _start
_start:
    ldr r1, =a
    ldr r1, [r1]
    ldr r2, =b
    ldr r2, [r2]
    add r1, r1, r2
    ldr r2, =c
    str r1, [r2]
    mov r7, #1
    svc #0
.end
```

At the bottom of the terminal window, there is a menu bar with "File Edit Tabs Help" and a status bar with "GNU nano 2.7.4 File: second. Modified". Below the status bar, there is a menu bar with keyboard shortcuts:
 ^G Get Help ^O Write Out ^W Where Is ^K Cut Text
 ^X Exit ^R Read File ^\ Replace ^U Uncut Text

To Assembly the file.

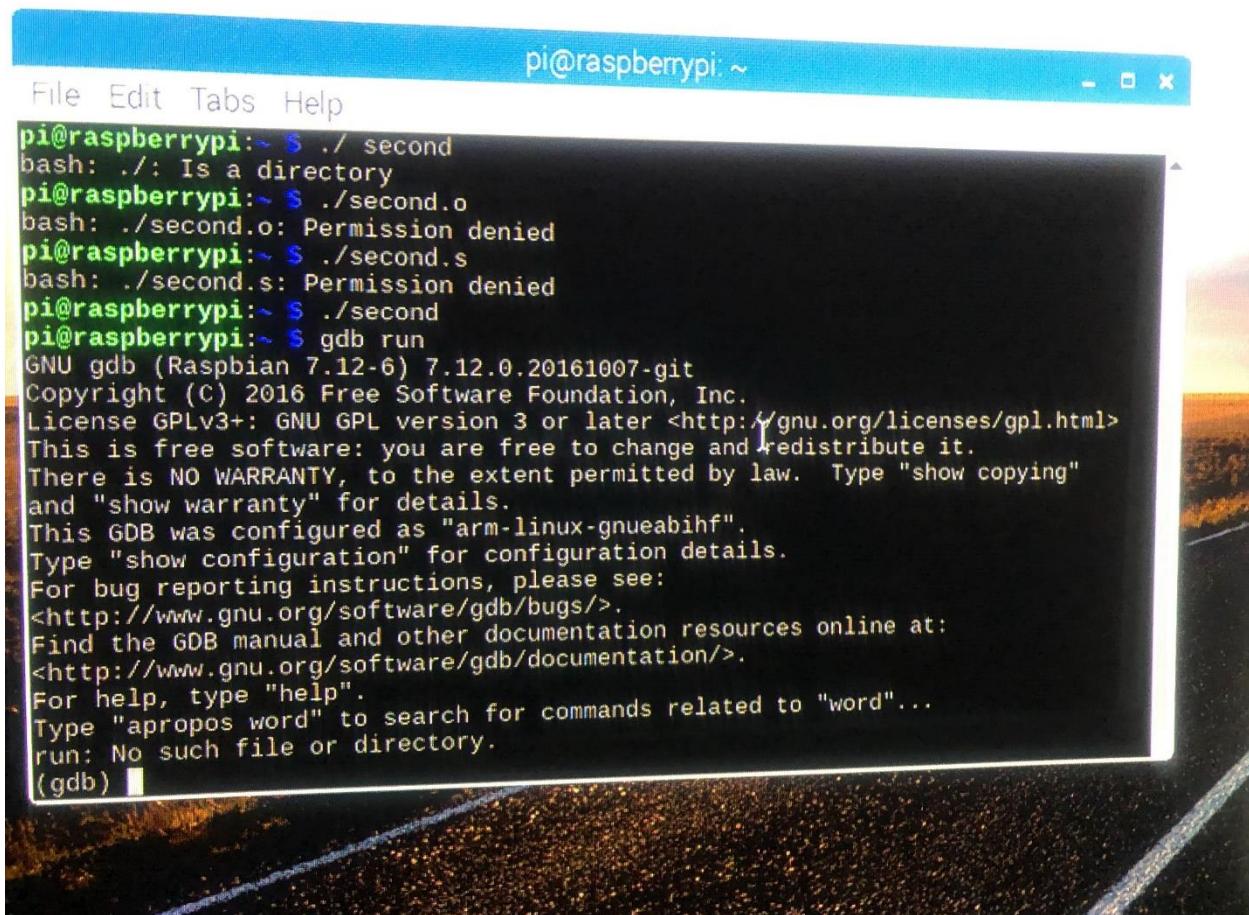
```
pi@raspberrypi:~ $ as -o second.o second.s
pi@raspberrypi:~ $ ld -o second second.o
pi@raspberrypi:~ $ ./second.o
bash: ./: Is a directory
pi@raspberrypi:~ $ ./second
bash: ./: Is a directory
pi@raspberrypi:~ $
```

Linked the program. Then trying to run it. error message bash: ./: is a directory



```
pi@raspberrypi:~ $ as -o second.o second.s
pi@raspberrypi:~ $ ld -o second second.o
pi@raspberrypi:~ $ ./second.o
bash: ./: Is a directory
pi@raspberrypi:~ $ ./second
bash: ./: Is a directory
pi@raspberrypi:~ $
```

Fixed and starting debugging process.



```
File Edit Tabs Help
pi@raspberrypi:~ $ ./ second
bash: ./: Is a directory
pi@raspberrypi:~ $ ./second.o
bash: ./second.o: Permission denied
pi@raspberrypi:~ $ ./second.s
bash: ./second.s: Permission denied
pi@raspberrypi:~ $ ./second
pi@raspberrypi:~ $ gdb run
GNU gdb (Raspbian 7.12-6) 7.12.0.20161007-git
Copyright (C) 2016 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-gnueabihf".
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"...
run: No such file or directory.
(gdb) [redacted]
```

Here we have listed the symbols from the object file

Here we are setting a break point for line 15 of the program.

```

pi@raspberrypi: ~
File Edit Tabs Help
2017-07-02-212258 1824x984 scrot.png 2019-02-15-1900114_1680x1050_scrot.png 2019-02-15-191156_1680x1050_scrot.png Music nukit Scratch second.o Templates
pi@raspberrypi: ~ $ id -o second.second.o
id: unrecognized option `--o'
id: use the --help option for usage information
pi@raspberrypi: ~ $ id -o second.second.o
id: unrecognized option `--o'
id: use the --help option for usage information
pi@raspberrypi: ~ $ id -o second.second.o
2017-07-02-212258 1824x984 scrot.png 2019-02-15-180956_1680x1050_scrot.png 2019-02-15-190551_1680x1050_scrot.png Desktop Pictures second.o Videos
2019-02-15-175419_1680x1050_scrot.png 2019-02-15-190702_1680x1050_scrot.png 2019-02-15-190755_1680x1050_scrot.png Documents Public second.s
pi@raspberrypi: ~ $ id -o second.second.o
id: unrecognized option `--o'
id: use the --help option for usage information
pi@raspberrypi: ~ $ id -o second.second.o
2017-07-02-212258 1824x984 scrot.png 2019-02-15-180956_1680x1050_scrot.png 2019-02-15-190551_1680x1050_scrot.png Desktop Pictures second.o Videos
2019-02-15-175419_1680x1050_scrot.png 2019-02-15-185702_1680x1050_scrot.png 2019-02-15-190755_1680x1050_scrot.png Documents Public second.s
2019-02-15-175634_1680x1050_scrot.png 2019-02-15-185816_1680x1050_scrot.png 2019-02-15-191100_1680x1050_scrot.png Downloads python_games spmd2
2019-02-15-180956_1680x1050_scrot.png 2019-02-15-190114_1680x1050_scrot.png 2019-02-15-191156_1680x1050_scrot.png Music Scratch spmd2.c
2019-02-15-180943_1680x1050_scrot.png 2019-02-15-190231_1680x1050_scrot.png 2019-02-15-191403_1680x1050_scrot.png nukit second Templates
pi@raspberrypi: ~ $ ./second
pi@raspberrypi: ~ $ gdb second
GNU gdb (Raspbian 7.7.1+dfsg-5+rpi1) 7.7.1
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3.0+: 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-gnueabihf".
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/>.
Type "help" to get help.
Type "apropos word" to search for commands related to "word"...
Reading symbols from second...done.
(gdb) list
1    @ second program: c = a + b
2    .section .text
3    .global _start
4    _start:
5        ldr r1, =a
6        ldr r2, =b
7        add r3, r1, r2
8        mov r0, r3
9        .section .data
10       .globl _start
11       _start:
12           ldr r1, =a
13           ldr r2, =b
14           add r3, r1, r2
15           mov r0, r3
16       .section .text
17       .globl _start
18       _start:
19           ldr r1, =a
20           ldr r2, =b
21           add r3, r1, r2
22           mov r0, r3
23       .section .data
24       .globl _start
25       _start:
26           ldr r1, =a
27           ldr r2, =b
28           add r3, r1, r2
29           mov r0, r3
30       .section .text
31       .globl _start
32       _start:
33           ldr r1, =a
34           ldr r2, =b
35           add r3, r1, r2
36           mov r0, r3
37       .section .data
38       .globl _start
39       _start:
40           ldr r1, =a
41           ldr r2, =b
42           add r3, r1, r2
43           mov r0, r3
44       .section .text
45       .globl _start
46       _start:
47           ldr r1, =a
48           ldr r2, =b
49           add r3, r1, r2
50           mov r0, r3
51       .section .data
52       .globl _start
53       _start:
54           ldr r1, =a
55           ldr r2, =b
56           add r3, r1, r2
57           mov r0, r3
58       .section .text
59       .globl _start
60       _start:
61           ldr r1, =a
62           ldr r2, =b
63           add r3, r1, r2
64           mov r0, r3
65       .section .data
66       .globl _start
67       _start:
68           ldr r1, =a
69           ldr r2, =b
70           add r3, r1, r2
71           mov r0, r3
72       .section .text
73       .globl _start
74       _start:
75           ldr r1, =a
76           ldr r2, =b
77           add r3, r1, r2
78           mov r0, r3
79       .section .data
80       .globl _start
81       _start:
82           ldr r1, =a
83           ldr r2, =b
84           add r3, r1, r2
85           mov r0, r3
86       .section .text
87       .globl _start
88       _start:
89           ldr r1, =a
90           ldr r2, =b
91           add r3, r1, r2
92           mov r0, r3
93       .section .data
94       .globl _start
95       _start:
96           ldr r1, =a
97           ldr r2, =b
98           add r3, r1, r2
99           mov r0, r3
100      .section .text
101      .globl _start
102      _start:
103          ldr r1, =a
104          ldr r2, =b
105          add r3, r1, r2
106          mov r0, r3
107      .section .data
108      .globl _start
109      _start:
110          ldr r1, =a
111          ldr r2, =b
112          add r3, r1, r2
113          mov r0, r3
114      .section .text
115      .globl _start
116      _start:
117          ldr r1, =a
118          ldr r2, =b
119          add r3, r1, r2
120          mov r0, r3
121      .section .data
122      .globl _start
123      _start:
124          ldr r1, =a
125          ldr r2, =b
126          add r3, r1, r2
127          mov r0, r3
128      .section .text
129      .globl _start
130      _start:
131          ldr r1, =a
132          ldr r2, =b
133          add r3, r1, r2
134          mov r0, r3
135      .section .data
136      .globl _start
137      _start:
138          ldr r1, =a
139          ldr r2, =b
140          add r3, r1, r2
141          mov r0, r3
142      .section .text
143      .globl _start
144      _start:
145          ldr r1, =a
146          ldr r2, =b
147          add r3, r1, r2
148          mov r0, r3
149      .section .data
150      .globl _start
151      _start:
152          ldr r1, =a
153          ldr r2, =b
154          add r3, r1, r2
155          mov r0, r3
156      .section .text
157      .globl _start
158      _start:
159          ldr r1, =a
160          ldr r2, =b
161          add r3, r1, r2
162          mov r0, r3
163      .section .data
164      .globl _start
165      _start:
166          ldr r1, =a
167          ldr r2, =b
168          add r3, r1, r2
169          mov r0, r3
170      .section .text
171      .globl _start
172      _start:
173          ldr r1, =a
174          ldr r2, =b
175          add r3, r1, r2
176          mov r0, r3
177      .section .data
178      .globl _start
179      _start:
180          ldr r1, =a
181          ldr r2, =b
182          add r3, r1, r2
183          mov r0, r3
184      .section .text
185      .globl _start
186      _start:
187          ldr r1, =a
188          ldr r2, =b
189          add r3, r1, r2
190          mov r0, r3
191      .section .data
192      .globl _start
193      _start:
194          ldr r1, =a
195          ldr r2, =b
196          add r3, r1, r2
197          mov r0, r3
198      .section .text
199      .globl _start
200      _start:
201          ldr r1, =a
202          ldr r2, =b
203          add r3, r1, r2
204          mov r0, r3
205      .section .data
206      .globl _start
207      _start:
208          ldr r1, =a
209          ldr r2, =b
210          add r3, r1, r2
211          mov r0, r3
212      .section .text
213      .globl _start
214      _start:
215          ldr r1, =a
216          ldr r2, =b
217          add r3, r1, r2
218          mov r0, r3
219      .section .data
220      .globl _start
221      _start:
222          ldr r1, =a
223          ldr r2, =b
224          add r3, r1, r2
225          mov r0, r3
226      .section .text
227      .globl _start
228      _start:
229          ldr r1, =a
230          ldr r2, =b
231          add r3, r1, r2
232          mov r0, r3
233      .section .data
234      .globl _start
235      _start:
236          ldr r1, =a
237          ldr r2, =b
238          add r3, r1, r2
239          mov r0, r3
240      .section .text
241      .globl _start
242      _start:
243          ldr r1, =a
244          ldr r2, =b
245          add r3, r1, r2
246          mov r0, r3
247      .section .data
248      .globl _start
249      _start:
250          ldr r1, =a
251          ldr r2, =b
252          add r3, r1, r2
253          mov r0, r3
254      .section .text
255      .globl _start
256      _start:
257          ldr r1, =a
258          ldr r2, =b
259          add r3, r1, r2
260          mov r0, r3
261      .section .data
262      .globl _start
263      _start:
264          ldr r1, =a
265          ldr r2, =b
266          add r3, r1, r2
267          mov r0, r3
268      .section .text
269      .globl _start
270      _start:
271          ldr r1, =a
272          ldr r2, =b
273          add r3, r1, r2
274          mov r0, r3
275      .section .data
276      .globl _start
277      _start:
278          ldr r1, =a
279          ldr r2, =b
280          add r3, r1, r2
281          mov r0, r3
282      .section .text
283      .globl _start
284      _start:
285          ldr r1, =a
286          ldr r2, =b
287          add r3, r1, r2
288          mov r0, r3
289      .section .data
290      .globl _start
291      _start:
292          ldr r1, =a
293          ldr r2, =b
294          add r3, r1, r2
295          mov r0, r3
296      .section .text
297      .globl _start
298      _start:
299          ldr r1, =a
300          ldr r2, =b
301          add r3, r1, r2
302          mov r0, r3
303      .section .data
304      .globl _start
305      _start:
306          ldr r1, =a
307          ldr r2, =b
308          add r3, r1, r2
309          mov r0, r3
310      .section .text
311      .globl _start
312      _start:
313          ldr r1, =a
314          ldr r2, =b
315          add r3, r1, r2
316          mov r0, r3
317      .section .data
318      .globl _start
319      _start:
320          ldr r1, =a
321          ldr r2, =b
322          add r3, r1, r2
323          mov r0, r3
324      .section .text
325      .globl _start
326      _start:
327          ldr r1, =a
328          ldr r2, =b
329          add r3, r1, r2
330          mov r0, r3
331      .section .data
332      .globl _start
333      _start:
334          ldr r1, =a
335          ldr r2, =b
336          add r3, r1, r2
337          mov r0, r3
338      .section .text
339      .globl _start
340      _start:
341          ldr r1, =a
342          ldr r2, =b
343          add r3, r1, r2
344          mov r0, r3
345      .section .data
346      .globl _start
347      _start:
348          ldr r1, =a
349          ldr r2, =b
350          add r3, r1, r2
351          mov r0, r3
352      .section .text
353      .globl _start
354      _start:
355          ldr r1, =a
356          ldr r2, =b
357          add r3, r1, r2
358          mov r0, r3
359      .section .data
360      .globl _start
361      _start:
362          ldr r1, =a
363          ldr r2, =b
364          add r3, r1, r2
365          mov r0, r3
366      .section .text
367      .globl _start
368      _start:
369          ldr r1, =a
370          ldr r2, =b
371          add r3, r1, r2
372          mov r0, r3
373      .section .data
374      .globl _start
375      _start:
376          ldr r1, =a
377          ldr r2, =b
378          add r3, r1, r2
379          mov r0, r3
380      .section .text
381      .globl _start
382      _start:
383          ldr r1, =a
384          ldr r2, =b
385          add r3, r1, r2
386          mov r0, r3
387      .section .data
388      .globl _start
389      _start:
390          ldr r1, =a
391          ldr r2, =b
392          add r3, r1, r2
393          mov r0, r3
394      .section .text
395      .globl _start
396      _start:
397          ldr r1, =a
398          ldr r2, =b
399          add r3, r1, r2
400          mov r0, r3
401      .section .data
402      .globl _start
403      _start:
404          ldr r1, =a
405          ldr r2, =b
406          add r3, r1, r2
407          mov r0, r3
408      .section .text
409      .globl _start
410      _start:
411          ldr r1, =a
412          ldr r2, =b
413          add r3, r1, r2
414          mov r0, r3
415      .section .data
416      .globl _start
417      _start:
418          ldr r1, =a
419          ldr r2, =b
420          add r3, r1, r2
421          mov r0, r3
422      .section .text
423      .globl _start
424      _start:
425          ldr r1, =a
426          ldr r2, =b
427          add r3, r1, r2
428          mov r0, r3
429      .section .data
430      .globl _start
431      _start:
432          ldr r1, =a
433          ldr r2, =b
434          add r3, r1, r2
435          mov r0, r3
436      .section .text
437      .globl _start
438      _start:
439          ldr r1, =a
440          ldr r2, =b
441          add r3, r1, r2
442          mov r0, r3
443      .section .data
444      .globl _start
445      _start:
446          ldr r1, =a
447          ldr r2, =b
448          add r3, r1, r2
449          mov r0, r3
450      .section .text
451      .globl _start
452      _start:
453          ldr r1, =a
454          ldr r2, =b
455          add r3, r1, r2
456          mov r0, r3
457      .section .data
458      .globl _start
459      _start:
460          ldr r1, =a
461          ldr r2, =b
462          add r3, r1, r2
463          mov r0, r3
464      .section .text
465      .globl _start
466      _start:
467          ldr r1, =a
468          ldr r2, =b
469          add r3, r1, r2
470          mov r0, r3
471      .section .data
472      .globl _start
473      _start:
474          ldr r1, =a
475          ldr r2, =b
476          add r3, r1, r2
477          mov r0, r3
478      .section .data
479      .globl _start
480      _start:
481          ldr r1, =a
482          ldr r2, =b
483          add r3, r1, r2
484          mov r0, r3
485      .section .text
486      .globl _start
487      _start:
488          ldr r1, =a
489          ldr r2, =b
490          add r3, r1, r2
491          mov r0, r3
492      .section .data
493      .globl _start
494      _start:
495          ldr r1, =a
496          ldr r2, =b
497          add r3, r1, r2
498          mov r0, r3
499      .section .data
500      .globl _start
501      _start:
502          ldr r1, =a
503          ldr r2, =b
504          add r3, r1, r2
505          mov r0, r3
506      .section .text
507      .globl _start
508      _start:
509          ldr r1, =a
510          ldr r2, =b
511          add r3, r1, r2
512          mov r0, r3
513      .section .data
514      .globl _start
515      _start:
516          ldr r1, =a
517          ldr r2, =b
518          add r3, r1, r2
519          mov r0, r3
520      .section .data
521      .globl _start
522      _start:
523          ldr r1, =a
524          ldr r2, =b
525          add r3, r1, r2
526          mov r0, r3
527      .section .text
528      .globl _start
529      _start:
530          ldr r1, =a
531          ldr r2, =b
532          add r3, r1, r2
533          mov r0, r3
534      .section .data
535      .globl _start
536      _start:
537          ldr r1, =a
538          ldr r2, =b
539          add r3, r1, r2
540          mov r0, r3
541      .section .text
542      .globl _start
543      _start:
544          ldr r1, =a
545          ldr r2, =b
546          add r3, r1, r2
547          mov r0, r3
548      .section .data
549      .globl _start
550      _start:
551          ldr r1, =a
552          ldr r2, =b
553          add r3, r1, r2
554          mov r0, r3
555      .section .text
556      .globl _start
557      _start:
558          ldr r1, =a
559          ldr r2, =b
560          add r3, r1, r2
561          mov r0, r3
562      .section .data
563      .globl _start
564      _start:
565          ldr r1, =a
566          ldr r2, =b
567          add r3, r1, r2
568          mov r0, r3
569      .section .data
570      .globl _start
571      _start:
572          ldr r1, =a
573          ldr r2, =b
574          add r3, r1, r2
575          mov r0, r3
576      .section .text
577      .globl _start
578      _start:
579          ldr r1, =a
580          ldr r2, =b
581          add r3, r1, r2
582          mov r0, r3
583      .section .data
584      .globl _start
585      _start:
586          ldr r1, =a
587          ldr r2, =b
588          add r3, r1, r2
589          mov r0, r3
590      .section .text
591      .globl _start
592      _start:
593          ldr r1, =a
594          ldr r2, =b
595          add r3, r1, r2
596          mov r0, r3
597      .section .data
598      .globl _start
599      _start:
599          ldr r1, =a
600          ldr r2, =b
601          add r3, r1, r2
602          mov r0, r3
603      .section .text
604      .globl _start
605      _start:
606          ldr r1, =a
607          ldr r2, =b
608          add r3, r1, r2
609          mov r0, r3
610      .section .data
611      .globl _start
612      _start:
613          ldr r1, =a
614          ldr r2, =b
615          add r3, r1, r2
616          mov r0, r3
617      .section .data
618      .globl _start
619      _start:
620          ldr r1, =a
621          ldr r2, =b
622          add r3, r1, r2
623          mov r0, r3
624      .section .text
625      .globl _start
626      _start:
627          ldr r1, =a
628          ldr r2, =b
629          add r3, r1, r2
630          mov r0, r3
631      .section .data
632      .globl _start
633      _start:
634          ldr r1, =a
635          ldr r2, =b
636          add r3, r1, r2
637          mov r0, r3
638      .section .text
639      .globl _start
640      _start:
641          ldr r1, =a
642          ldr r2, =b
643          add r3, r1, r2
644          mov r0, r3
645      .section .data
646      .globl _start
647      _start:
648          ldr r1, =a
649          ldr r2, =b
650          add r3, r1, r2
651          mov r0, r3
652      .section .text
653      .globl _start
654      _start:
655          ldr r1, =a
656          ldr r2, =b
657          add r3, r1, r2
658          mov r0, r3
659      .section .data
660      .globl _start
661      _start:
662          ldr r1, =a
663          ldr r2, =b
664          add r3, r1, r2
665          mov r0, r3
666      .section .text
667      .globl _start
668      _start:
669          ldr r1, =a
670          ldr r2, =b
671          add r3, r1, r2
672          mov r0, r3
673      .section .data
674      .globl _start
675      _start:
676          ldr r1, =a
677          ldr r2, =b
678          add r3, r1, r2
679          mov r0, r3
680      .section .text
681      .globl _start
682      _start:
683          ldr r1, =a
684          ldr r2, =b
685          add r3, r1, r2
686          mov r0, r3
687      .section .data
688      .globl _start
689      _start:
690          ldr r1, =a
691          ldr r2, =b
692          add r3, r1, r2
693          mov r0, r3
694      .section .text
695      .globl _start
696      _start:
697          ldr r1, =a
698          ldr r2, =b
699          add r3, r1, r2
700          mov r0, r3
701      .section .data
702      .globl _start
703      _start:
704          ldr r1, =a
705          ldr r2, =b
706          add r3, r1, r2
707          mov r0, r3
708      .section .text
709      .globl _start
710      _start:
711          ldr r1, =a
712          ldr r2, =b
713          add r3, r1, r2
714          mov r0, r3
715      .section .data
716      .globl _start
717      _start:
718          ldr r1, =a
719          ldr r2, =b
720          add r3, r1, r2
721          mov r0, r3
722      .section .text
723      .globl _start
724      _start:
725          ldr r1, =a
726          ldr r2, =b
727          add r3, r1, r2
728          mov r0, r3
729      .section .data
730      .globl _start
731      _start:
732          ldr r1, =a
733          ldr r2, =b
734          add r3, r1, r2
735          mov r0, r3
736      .section .text
737      .globl _start
738      _start:
739          ldr r1, =a
740          ldr r2, =b
741          add r3, r1, r2
742          mov r0, r3
743      .section .data
744      .globl _start
745      _start:
746          ldr r1, =a
747          ldr r2, =b
748          add r3, r1, r2
749          mov r0, r3
750      .section .text
751      .globl _start
752      _start:
753          ldr r1, =a
754          ldr r2, =b
755          add r3, r1, r2
756          mov r0, r3
757      .section .data
758      .globl _start
759      _start:
760          ldr r1, =a
761          ldr r2, =b
762          add r3, r1, r2
763          mov r0, r3
764      .section .text
765      .globl _start
766      _start:
767          ldr r1, =a
768          ldr r2, =b
769          add r3, r1, r2
770          mov r0, r3
771      .section .data
772      .globl _start
773      _start:
774          ldr r1, =a
775          ldr r2, =b
776          add r3, r1, r2
777          mov r0, r3
778      .section .text
779      .globl _start
780      _start:
781          ldr r1, =a
782          ldr r2, =b
783          add r3, r1, r2
784          mov r0, r3
785      .section .data
786      .globl _start
787      _start:
788          ldr r1, =a
789          ldr r2, =b
790          add r3, r1, r2
791          mov r0, r3
792      .section .text
793      .globl _start
794      _start:
795          ldr r1, =a
796          ldr r2, =b
797          add r3, r1, r2
798          mov r0, r3
799      .section .data
800      .globl _start
801      _start:
802          ldr r1, =a
803          ldr r2, =b
804          add r3, r1, r2
805          mov r0, r3
806      .section .text
807      .globl _start
808      _start:
809          ldr r1, =a
810          ldr r2, =b
811          add r3, r1, r2
812          mov r0, r3
813      .section .data
814      .globl _start
815      _start:
816          ldr r1, =a
817          ldr r2, =b
818          add r3, r1, r2
819          mov r0, r3
820      .section .text
821      .globl _start
822      _start:
823          ldr r1, =a
824          ldr r2, =b
825          add r3, r1, r2
826          mov r0, r3
827      .section .data
828      .globl _start
829      _start:
830          ldr r1, =a
831          ldr r2, =b
832          add r3, r1, r2
833          mov r0, r3
834      .section .text
835      .globl _start
836      _start:
837          ldr r1, =a
838          ldr r2, =b
839          add r3, r1, r2
840          mov r0, r3
841      .section .data
842      .globl _start
843      _start:
844          ldr r1, =a
845          ldr r2, =b
846          add r3, r1, r2
847          mov r0, r3
848      .section .text
849      .globl _start
850      _start:
851          ldr r1, =a
852          ldr r2, =b
853          add r3, r1, r2
854          mov r0, r3
855      .section .data
856      .globl _start
857      _start:
858          ldr r1, =a
859          ldr r2, =b
860          add r3, r1, r2
861          mov r0, r3
862      .section .text
863      .globl _start
864      _start:
865          ldr r1, =a
866          ldr r2, =b
867          add r3, r1, r2
868          mov r0, r3
869      .section .data
870      .globl _start
871      _start:
872          ldr r1, =a
873          ldr r2, =b
874          add r3, r1, r2
875          mov r0, r3
876      .section .text
877      .globl _start
878      _start:
879          ldr r1, =a
880          ldr r2, =b
881          add r3, r1, r2
882          mov r0, r3
883      .section .data
884      .globl _start
885      _start:
886          ldr r1, =a
887          ldr r2, =b
888          add r3, r1, r2
889          mov r0, r3
890      .section .text
891      .globl _start
892      _start:
893          ldr r1, =a
894          ldr r2, =b
895          add r3, r1, r2
896          mov r0, r3
897      .section .data
898      .globl _start
899      _start:
900          ldr r1, =a
901          ldr r2, =b
902          add r3, r1, r2
903          mov r0, r3
904      .section .text
905      .globl _start
906      _start:
907          ldr r1, =a
908          ldr r2, =b
909          add r3, r1, r2
910          mov r0, r3
911      .section .data
912      .globl _start
913      _start:
914          ldr r1, =a
915          ldr r2, =b
916          add r3, r1, r2
917          mov r0, r3
918      .section .data
919      .globl _start
920      _start:
921          ldr r1, =a
922          ldr r2, =b
923          add r3, r1, r2
924          mov r0, r3
925      .section .text
926      .globl _start
927      _start:
928          ldr r1, =a
929          ldr r2, =b
930          add r3, r1, r2
931          mov r0, r3
932      .section .data
933      .globl _start
934      _start:
935          ldr r1, =a
936          ldr r2, =b
937          add r3, r1, r2
938          mov r0, r3
939      .section .text

```

Here are stepping into instructions, the first instruction on line 15 has yet to be executed

```
pi@raspberrypi: ~
```

File Edt Tabs Help

```
2017-07-02-212258_1624x904_screenshot.png 2019-02-15-180955_1680x1050_screenshot.png 2019-02-15-190551_1680x1050_screenshot.png Desktop Pictures second.o Videos
2019-02-15-175419_1680x1050_screenshot.png 2019-02-15-185702_1680x1050_screenshot.png 2019-02-15-190755_1680x1050_screenshot.png Documents Public second.s
2019-02-15-175634_1680x1050_screenshot.png 2019-02-15-185816_1680x1050_screenshot.png 2019-02-15-191100_1680x1050_screenshot.png Downloads python_games spnd2
2019-02-15-180558_1680x1050_screenshot.png 2019-02-15-190114_1680x1050_screenshot.png 2019-02-15-191156_1680x1050_screenshot.png Music Scratch spnd2.c
2019-02-15-180943_1680x1050_screenshot.png 2019-02-15-190231_1680x1050_screenshot.png 2019-02-15-191403_1680x1050_screenshot.png nukkit second Templates
pi@raspberrypi: ~ $ ./second
pi@raspberrypi: ~ $ ./second.o second.o second.s
pi@raspberrypi: ~ $ ls
pi@raspberrypi: ~ $ ./ls
2017-07-02-212258_1624x904_screenshot.png 2019-02-15-180955_1680x1050_screenshot.png 2019-02-15-190551_1680x1050_screenshot.png Desktop Pictures second.o Videos
2019-02-15-175419_1680x1050_screenshot.png 2019-02-15-185702_1680x1050_screenshot.png 2019-02-15-190755_1680x1050_screenshot.png Documents Public second.s
2019-02-15-175634_1680x1050_screenshot.png 2019-02-15-185816_1680x1050_screenshot.png 2019-02-15-191100_1680x1050_screenshot.png Downloads python_games spnd2
2019-02-15-180558_1680x1050_screenshot.png 2019-02-15-190114_1680x1050_screenshot.png 2019-02-15-191156_1680x1050_screenshot.png Music Scratch spnd2.c
2019-02-15-180943_1680x1050_screenshot.png 2019-02-15-190231_1680x1050_screenshot.png 2019-02-15-191403_1680x1050_screenshot.png nukkit second Templates
pi@raspberrypi: ~ $ ldd ./second second.o
second.o: in function _start
  undefined reference to `__cxa_start_main'
second.o: in function __text_start()
  undefined reference to `__cxa_start_main'
pi@raspberrypi: ~ $ ld -o second second.o
pi@raspberrypi: ~ $ ls
2017-07-02-212258_1624x904_screenshot.png 2019-02-15-180955_1680x1050_screenshot.png 2019-02-15-190551_1680x1050_screenshot.png Desktop Pictures second.o Videos
2019-02-15-175419_1680x1050_screenshot.png 2019-02-15-185702_1680x1050_screenshot.png 2019-02-15-190755_1680x1050_screenshot.png Documents Public second.s
2019-02-15-175634_1680x1050_screenshot.png 2019-02-15-185816_1680x1050_screenshot.png 2019-02-15-191100_1680x1050_screenshot.png Downloads python_games spnd2
2019-02-15-180558_1680x1050_screenshot.png 2019-02-15-190114_1680x1050_screenshot.png 2019-02-15-191156_1680x1050_screenshot.png Music Scratch spnd2.c
2019-02-15-180943_1680x1050_screenshot.png 2019-02-15-190231_1680x1050_screenshot.png 2019-02-15-191403_1680x1050_screenshot.png nukkit second Templates
pi@raspberrypi: ~ $ ./second
pi@raspberrypi: ~ $ ./second.o
Copyright (C) 2014 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; for details see the COPYING file.
To extend the warranty for this software enter the warranty period with the command
  warranty_start [seconds]
This GDB was configured as "armlinux-gnueabihf".
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 second...done.
(gdb) list
1  .second program: c = a + b
2   .section .data
3   a: .word 2 @32-bit variable a in memory
4   b: .word 5 @32-bit variable b in memory
5   c: .word 0 @32-bit variable c in memory
6
7   .section .text
8   .globl _start
9   _start:
10    ldr r1, =@load the memory address of a into r1
(gdb) b 15
Breakpoint 1 at 0x10088: file second.s, line 15.
(gdb) run
Starting program: /home/pi/second

Breakpoint 1, _start () at second.s:15
15          ldr r2, =@load the memory address of c into r2
(gdb) stepi
16          str r1,[r2] @store r1 into memory c
(gdb) 
```

Here we can see info for line 15 the registers shows r1 = 7, r2 = 5

```
pi@raspberrypi: ~
File Edit Tabs Help
(gdb) info registers
r0          0x0      0
r1          0x0      0
r2          0x200ac  131244
r3          0x0      0
r4          0x0      0
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          0x7effff520 0x7effff520
lr          0x0      0
pc          0x1008c 0x1008c <_start+24>
cpsr        0x10      16
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/second

Breakpoint 1,  start () at second.s:15
15          ldr r2, =c @load the memory address of c into r2
(gdb) info line
Line 15 of `second.s' starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) x/3xw $10088
0x10088 <_start+20>: 0xe59f2010 0xe5821000 0xe3a07001
(gdb) next
16          str r1,[r2] @store r1 into memory c
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/second

Breakpoint 1, _start () at second.s:15
15          ldr r2, =c @load the memory address of c into r2
(gdb) info line
Line 15 of `second.s' starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) x/3xw $10088
0x10088 <_start+20>: 0xe59f2010 0xe5821000 0xe3a07001
(gdb) info registers
r0          0x0      0
r1          0x7      7
r2          0x5      5
r3          0x0      0
r4          0x0      0
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          0x7effff520 0x7effff520
lr          0x0      0
pc          0x10088 0x10088 <_start+20>
cpsr        0x10      16
(gdb) 
```

Next we see line 16. Here we can see that r3 is holding the memory location for c.

```
(gdb) info line
Line 15 of "second.s" starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) info line
Line 16 of "second.s" starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) next
16      str r1,[r2] @store r1 into memory c
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/second

Breakpoint 1, _start () at second.s:15
15      ldr r2, =@load the memory address of c into r2
(gdb) info line
Line 15 of "second.s" starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) x/3wx 0x10088
0x10088 <_start+20>: 0xe59f2010 0xe5821000 0xe3a07001
(gdb) next
16      str r1,[r2] @store r1 into memory c
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/second

Breakpoint 1, _start () at second.s:15
15      ldr r2, =@load the memory address of c into r2
(gdb) info line
Line 15 of "second.s" starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) x/3wx 0x10088
0x10088 <_start+20>: 0xe59f2010 0xe5821000 0xe3a07001
(gdb) info registers
r0      0x0      0
r1      0x7      7
r2      0x5      5
r3      0x0      0
r4      0x0      0
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10088 0x10088 <_start+20>
cpsr   0x10      16
(gdb) next
16      str r1,[r2] @store r1 into memory c
(gdb) info line
Line 16 of "second.s" starts at address 0x1008c <_start+24> and ends at 0x10090 <_start+28>.
(gdb) x/3wx 0x1008c
0x1008c <_start+24>: 0xe5821000 0xe3a07001 0xef000000
(gdb) info registers
r0      0x0      0
r1      0x7      7
r2      0x200ac 131244
r3      0x0      0
r4      0x0      0
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10090 0x10090 <_start+28>
cpsr   0x10      16
(gdb)
```

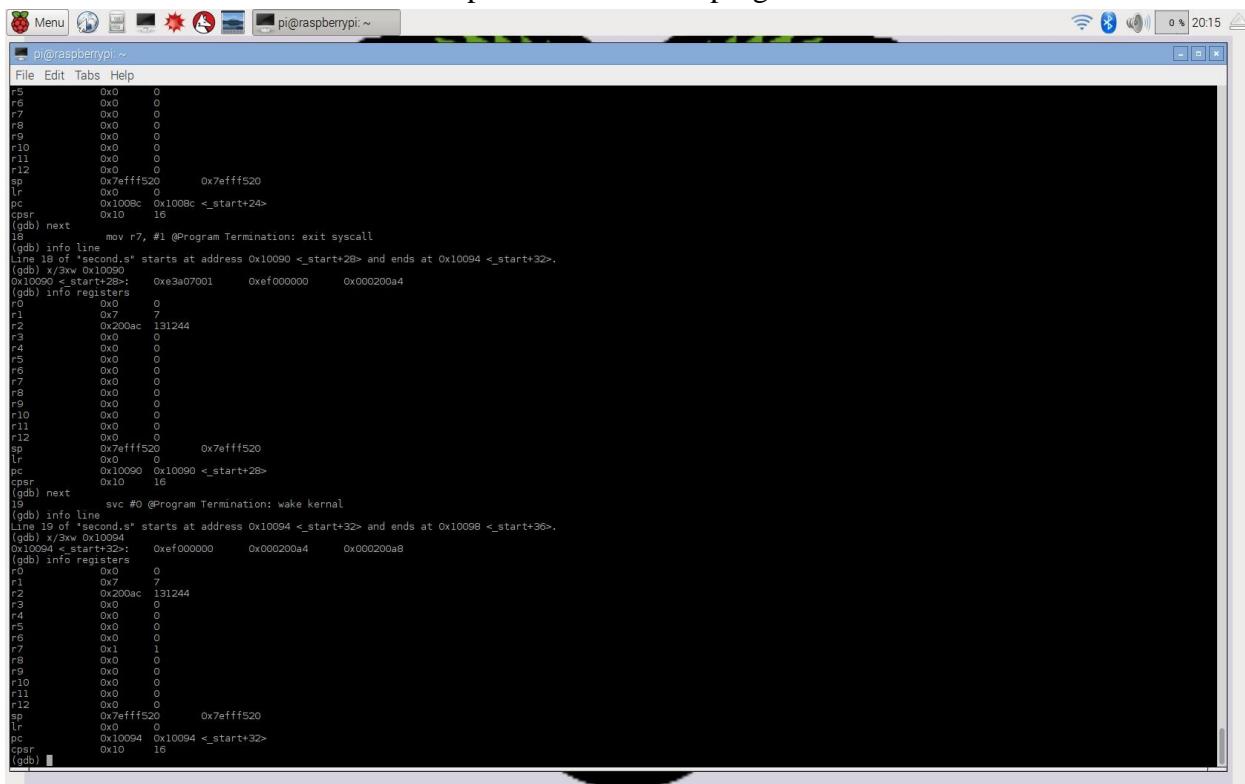
Below we can see that line 17 was not shown in the gdb debugger.

```
(gdb) info line
Line 19 of "second.s" starts at address 0x10094 <_start+32> and ends at 0x10098 <_start+36>.
(gdb) info line
Line 19 of "second.s" starts at address 0x10094 <_start+32> and ends at 0x10098 <_start+36>.
(gdb) info registers
r0      0x0      0
r1      0x7      7
r2      0x200ac 131244
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10094 0x10094 <_start+32>
cpsr   0x10      16
(gdb) next
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/second

Breakpoint 1, _start () at second.s:15
15      ldr r2, =@load the memory address of c into r2
(gdb) next
16      str r1,[r2] @store r1 into memory c
(gdb) next
17      mov r7, #1 @Program Termination: exit syscall
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/second

Breakpoint 1, _start () at second.s:15
15      ldr r2, =@load the memory address of c into r2
(gdb) stepi
16      str r1,[r2] @store r1 into memory c
(gdb) stepi
18      mov r7, #1 @Program Termination: exit syscall
(gdb)
```

Here below we can see that a 1 was put into r7 and the program exited to the kernel.



The screenshot shows a terminal window titled "pi@raspberrypi: ~" running on a Raspberry Pi. The window contains a GDB session. The command "info registers" is run twice, showing register values before and after a "next" step. In the second "info registers" output, r7 is shown with a value of 1, indicating it has been modified. The session ends with a "svc #0" command, which exits the program to the kernel.

```
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x1008c 0x1008c <_start+24>
cpsr    0x10    16
(gdb) next
18      mov r7, #1 @Program Termination: exit syscall
(gdb) info line
Line 18 of "second.s" starts at address 0x10090 <_start+28> and ends at 0x10094 <_start+32>.
(gdb) x/3w $0x10090
0x10090 <_start+28>: 0xe3a07001 0xef000000 0x000200a4
(gdb) info registers
r0      0x0      0
r1      0x7      7
r2      0x2000ac 0x1244
r3      0x0      0
r4      0x0      0
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10090 0x10090 <_start+28>
cpsr    0x10    16
(gdb) next
19      svc #0 @Program Termination: wake kernel
(gdb) info line
Line 19 of "second.s" starts at address 0x10094 <_start+32> and ends at 0x10098 <_start+36>.
(gdb) x/3w $0x10094
0x10094 <_start+32>: 0xef000000 0x000200a4 0x000200a8
(gdb) info registers
r0      0x0      0
r1      0x7      7
r2      0x2000ac 0x1244
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10094 0x10094 <_start+32>
cpsr    0x10    16
(gdb)
```

Writing the code on ARM assembly with registers part 2

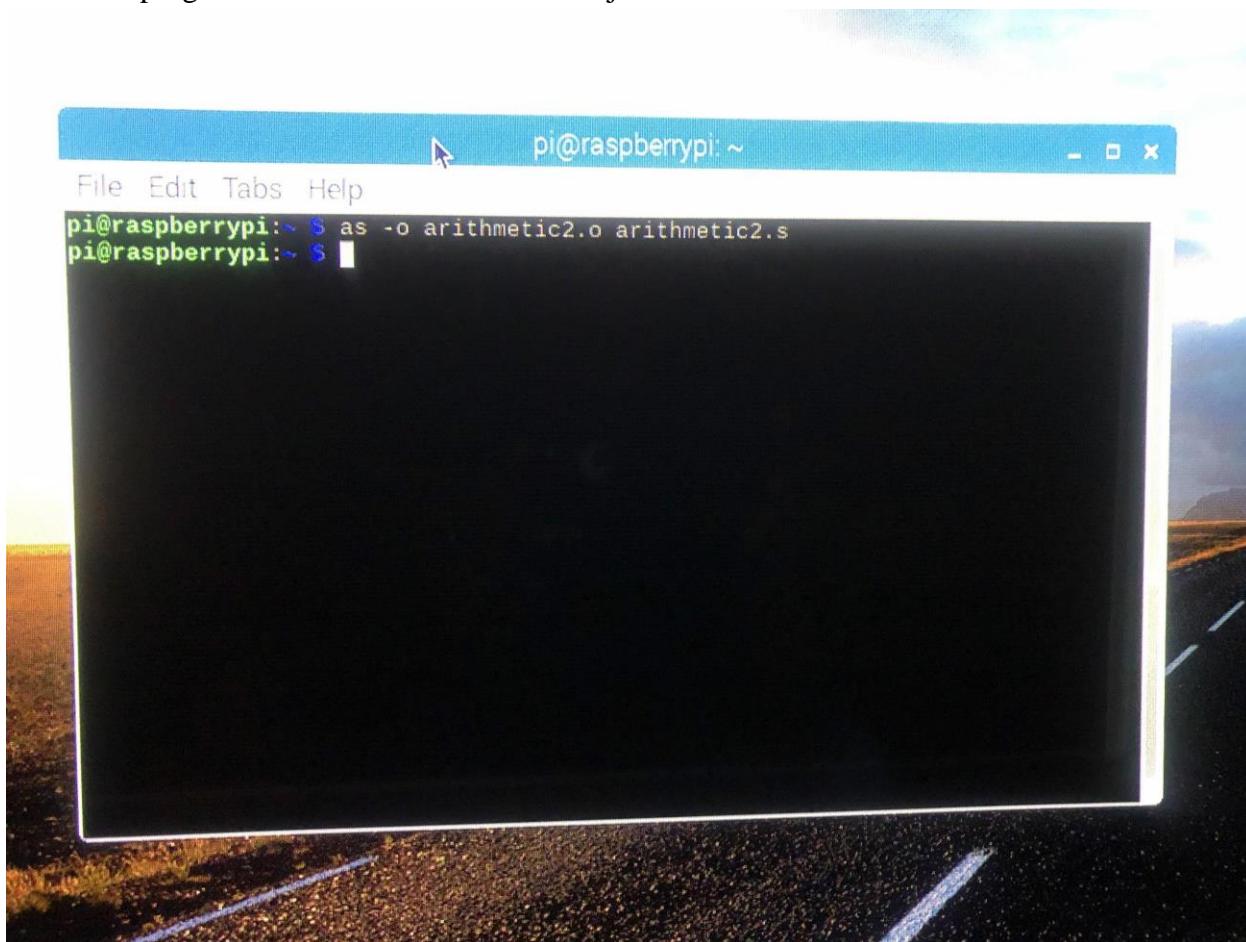
The screenshot shows a terminal window titled "pi@raspberrypi: ~" running the "GNU nano 2.7.4" editor. The file being edited is named "arithmetic2.s". The assembly code defines three words in the .data section and performs arithmetic operations in the .text section:

```
@ arithmetic2
.section .data
val1: .word 6
val2: .word 11
val3: .word 16

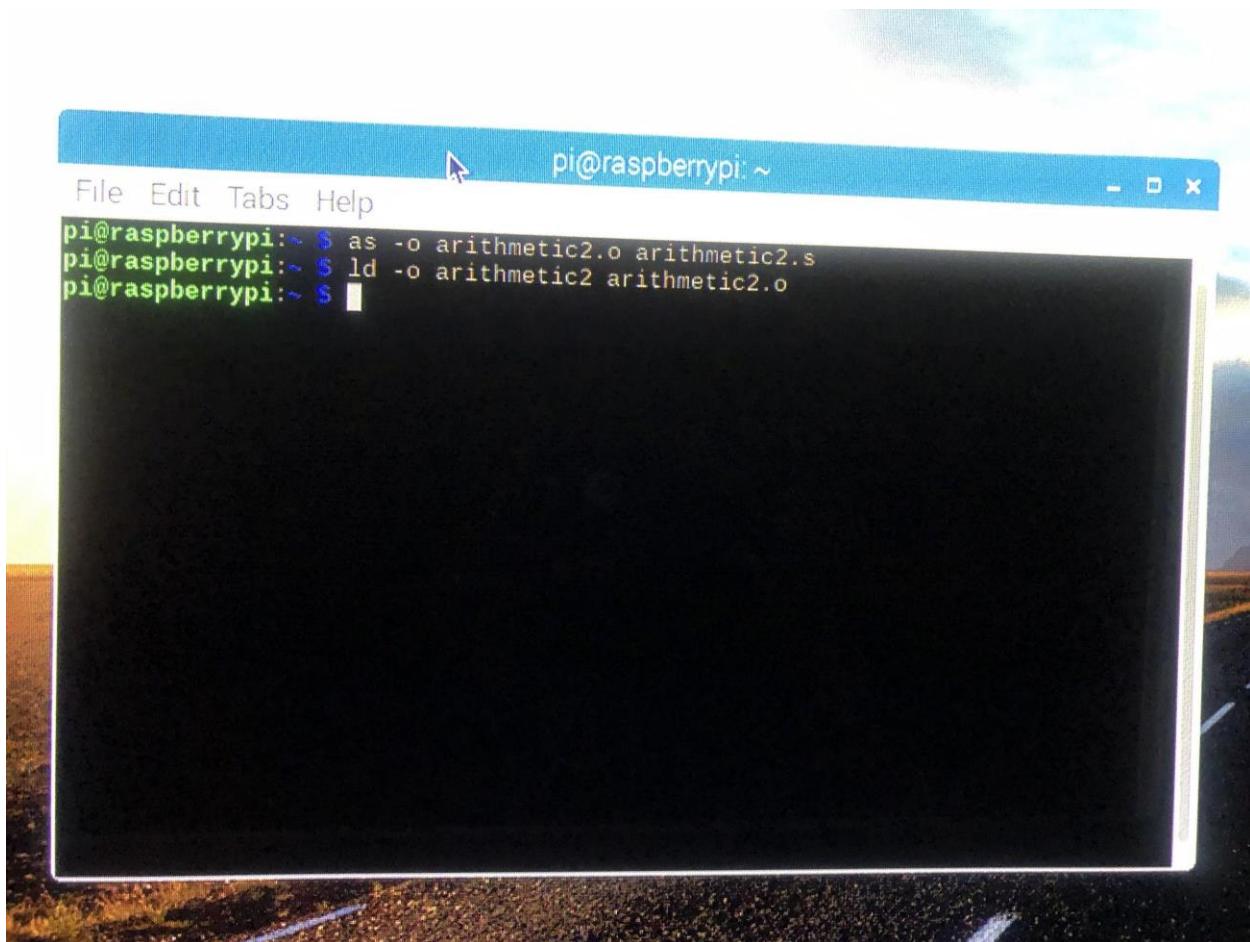
.section .text
.globl _start
_start:
    ldr r1, =val1
    ldr r1, [r1]
    ldr r2, =val2
    ldr r2, [r2]
    ldr r3, =val3
    ldr r3, [r3]
    add r2, #9
    add r2, r3
    sub r2, r1
```

The status bar at the bottom of the terminal window displays keyboard shortcuts for various functions like Get Help, Write Out, Where Is, Cut Text, Justify, Cur Pos, Exit, Read File, Replace, Uncut Text, To Spell, and Go To Line. It also indicates that 22 lines have been read.

Assemble program. arithmetic2.o create file object.

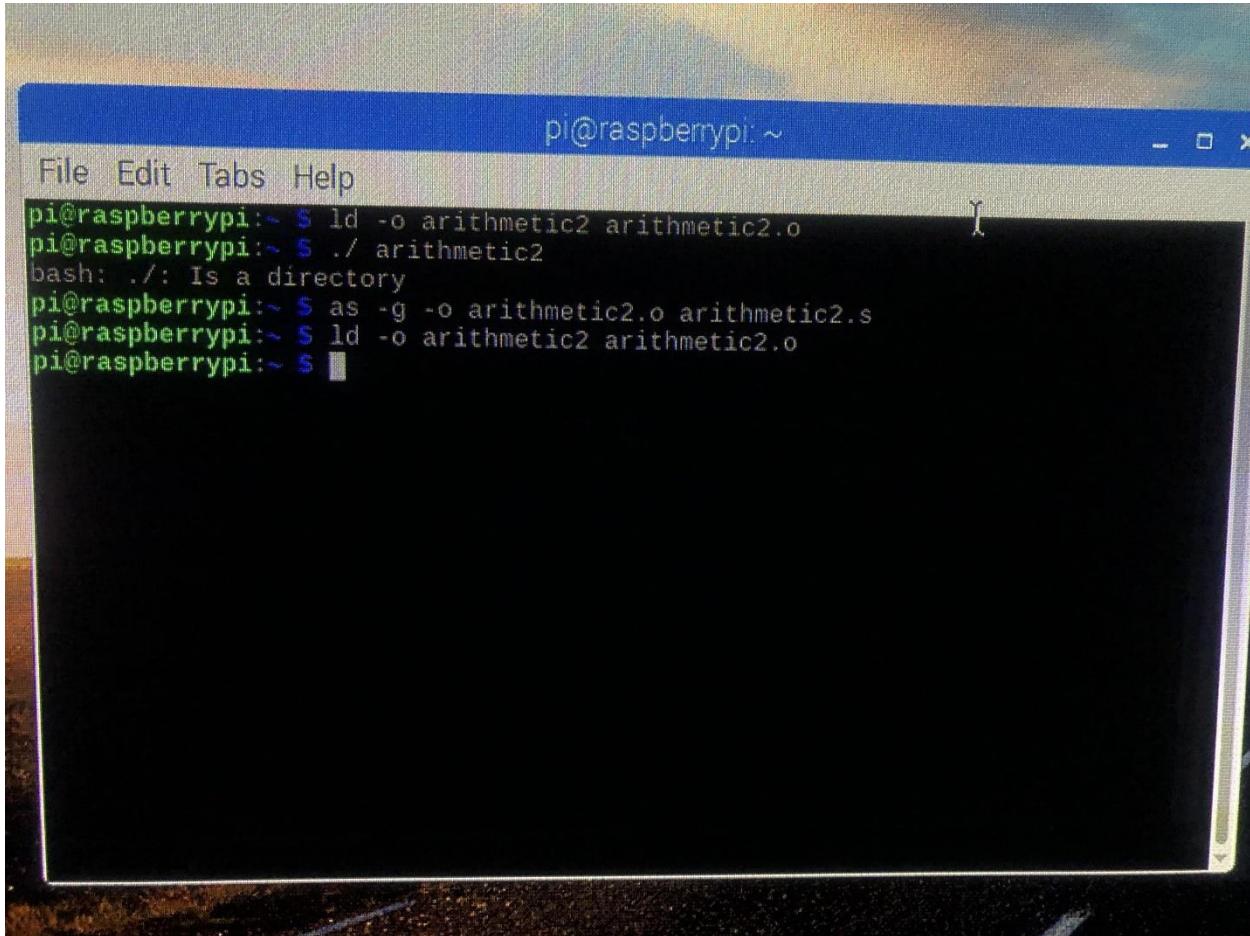


Link file in the second line of code.



```
File Edit Tabs Help
pi@raspberrypi:~ $ as -o arithmetic2.o arithmetic2.s
pi@raspberrypi:~ $ ld -o arithmetic2 arithmetic2.o
pi@raspberrypi:~ $
```

The second line of command trying to run the program, but there is an error message (bash: ./ is a directory). Then I try to debug by adding (-g) to third line of the program.



```
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi: ~ $ ld -o arithmetic2 arithmetic2.o
pi@raspberrypi: ~ $ ./ arithmetic2
bash: ./: Is a directory
pi@raspberrypi: ~ $ as -g -o arithmetic2.o arithmetic2.s
pi@raspberrypi: ~ $ ld -o arithmetic2 arithmetic2.o
pi@raspberrypi: ~ $
```

Found out the file did not end with svc #0 and .end which was causing run errors.

Next we use the list command.

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi: $ chmod --help
Usage: chmod [OPTION]... MODE FILE...
      chmod [OPTION]... --reference=FILE FILE...
      chmod [OPTION]... -reference=FILE FILE...
Change the mode of each FILE to MODE.
With --reference, change the mode of each FILE to that of FILE.
      -c, --changes          like verbose but report only when a change is made
      -f, --silent, --quiet    suppress most error messages
      -v, --verbose           output a diagnostic for every file processed
      --no-preserve-root      do not treat '/' specially (the default)
      --preserve-root         fail to operate recursively on '/'
      --reference=FILE        use FILE mode instead of MODE values
      -R, --recursive          change files and directories recursively
      --help                  display this help and exit
      --version               output version information and exit

Each MODE is of the form `[:ugo]*([+=][[:ugo]])+|[+=][0-7]+`.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
Full documentation at: <http://www.gnu.org/software/coreutils/chmod>
or available locally via: /usr/share/doc/coreutils/html/manual/chmod.html

pi@raspberrypi: $ chmod arithmetic2 xxx
chmod: invalid mode: `arithmetic2'
Try `chmod --help' for more information.
pi@raspberrypi: $ chmod arithmetic2 777
chmod: invalid mode: `777'
Try `chmod --help' for more information.
pi@raspberrypi: $ chmod 777 arithmetic2
pi@raspberrypi: $ ./arithmetic2
Illegal instruction
pi@raspberrypi: $ gdb arithmetic2
GNU gdb (Raspbian 7.7.1-dfsg-5+rpi1) 7.7.1
Copyright (C) 2014 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-gnueabihf'.
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 arithmetic2...done.
(gdb) list
Undefined command: "list". Try "help".
(gdb) list
1     @arithmetic_2
2         .section .data
3             val1: .word 6
4             val2: .word 11
5             val3: .word 16
6
7         .section .text
8         .globl _start
9         _start:
10            ldr r1, =val1
(gdb) b 15
Breakpoint 1 at 0x100088: file arithmetic2.s, line 15.
(gdb) 

```

Here we set a breakpoint

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi: $ chmod --help
Usage: chmod [OPTION]... MODE FILE...
      chmod [OPTION]... --reference=FILE FILE...
      chmod [OPTION]... -reference=FILE FILE...
Change the mode of each FILE to MODE.
With --reference, change the mode of each FILE to that of FILE.
      -c, --changes          like verbose but report only when a change is made
      -f, --silent, --quiet    suppress most error messages
      -v, --verbose           output a diagnostic for every file processed
      --no-preserve-root      do not treat '/' specially (the default)
      --preserve-root         fail to operate recursively on '/'
      --reference=FILE        use FILE mode instead of MODE values
      -R, --recursive          change files and directories recursively
      --help                  display this help and exit
      --version               output version information and exit

Each MODE is of the form `[:ugo]*([+=][[:ugo]])+|[+=][0-7]+`.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
Full documentation at: <http://www.gnu.org/software/coreutils/chmod>
or available locally via: /usr/share/doc/coreutils/html/manual/chmod.html

pi@raspberrypi: $ chmod arithmetic2 xxx
chmod: invalid mode: `arithmetic2'
Try `chmod --help' for more information.
pi@raspberrypi: $ chmod arithmetic2 777
chmod: invalid mode: `777'
Try `chmod --help' for more information.
pi@raspberrypi: $ chmod 777 arithmetic2
pi@raspberrypi: $ ./arithmetic2
Illegal instruction
pi@raspberrypi: $ gdb arithmetic2
GNU gdb (Raspbian 7.7.1-dfsg-5+rpi1) 7.7.1
Copyright (C) 2014 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-gnueabihf'.
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 arithmetic2...done.
(gdb) list
Undefined command: "list". Try "help".
(gdb) list
1     @arithmetic_2
2         .section .data
3             val1: .word 6
4             val2: .word 11
5             val3: .word 16
6
7         .section .text
8         .globl _start
9         _start:
10            ldr r1, =val1
(gdb) b 15
Breakpoint 1 at 0x100088: file arithmetic2.s, line 15.
(gdb) 

```

Next we see lines 15-19 executed and then the programs exits and give control to the kernal.
R3 loaded with value

```

File Edit Tabs Help
4      .val2: .word 11
5      .val3: .word 16
6
7      .section .text
8      .globl _start
9      _start:
10     ldr r1, = val1
11     b 15
Breakpoint 1 at 0x10088: file arithmetic2.s, line 15.
(gdb) run
Starting program: /home/pi/arithmetic2
Breakpoint 1, start () at arithmetic2.s:15
15    ldr r3, [r3]
(gdb) info line
Line 15 of `arithmetic2.s' starts at address 0x10088 <_start+20> and ends at 0x1008c <_start+24>.
(gdb) x/3w $1008c <_start+20>
0x1008c <_start+20>: 0xe5933000 0xe2822009 0xe0822003
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x11     11
r3      0x200aac 131244
r4      0x0      0
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      0x7effff520 0x7effff520
lr      0x0      0
pc      0x10088 0x10088 <_start+20>
cpsr   0x10      16
(gdb) stepi  add r2, #9
(gdb) info line
Line 16 of `arithmetic2.s' starts at address 0x1008c <_start+24> and ends at 0x10090 <_start+28>.
(gdb) x/3w $10090 <_start+24>
0x10090 <_start+24>: 0xe2822009 0xe0822003 0xe0422001
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x11     11
r3      0x10      16
r4      0x0      0
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      0x7effff520 0x7effff520
lr      0x0      0
pc      0x1008c 0x1008c <_start+24>
cpsr   0x10      16
(gdb)

```

Line 17 at bottom of picture is #9 added to r2

```

File Edit Tabs Help
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      0x7effff520 0x7effff520
lr      0x0      0
pc      0x10088 0x10088 <_start+20>
cpsr   0x10      16
(gdb) stepi  add r2, #9
(gdb) info line
Line 16 of `arithmetic2.s' starts at address 0x1008c <_start+24> and ends at 0x10090 <_start+28>.
(gdb) x/3w $10090 <_start+28>
0x10090 <_start+28>: 0xe2822009 0xe0822003 0xe0422001
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x11     11
r3      0x10      16
r4      0x0      0
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      0x7effff520 0x7effff520
lr      0x0      0
pc      0x10090 0x10090 <_start+28>
cpsr   0x10      16
(gdb) stepi  add r2, r3
(gdb) info line
Line 17 of `arithmetic2.s' starts at address 0x10090 <_start+28> and ends at 0x10094 <_start+32>.
(gdb) x/3w $10094 <_start+32>
0x10094 <_start+32>: 0xe0822003 0xe0422001 0x000200a4
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x14     20
r3      0x10      16
r4      0x0      0
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      0x7effff520 0x7effff520
lr      0x0      0
pc      0x10090 0x10090 <_start+28>
cpsr   0x10      16
(gdb)

```

Finally we see a subtraction of r1 from r2 which gives a final answer of 30

```

pi@raspberrypi:~ 
File Edit Tabs Help
19    mov r7, #1
(gdb) run
This program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pi/arithmetic2
Breakpoint 1, _start () at arithmetic2.s:15
15        ldr r3, [r3]
(gdb) next
16        add r2, #9
(gdb) next
17        add r2, r3
(gdb) next
18        sub r2, r1
(gdb) info line
Line 18 of `arithmetic2.s' starts at address 0x10094 <_start+32> and ends at 0x10098 <_start+36>.
(gdb) x/30x $r0-$r12
0x10094 <_start+32>: 0xe0422001 0xe3a07001 0xef000000
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x24     36
r3      0x10     16
r4      0x0      0
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10094 0x10094 <_start+32>
cpsr   0x10     16
(gdb) stepi
19    mov r7, #1
(gdb) info line
Line 19 of `arithmetic2.s' starts at address 0x10098 <_start+36> and ends at 0x1009c <_start+40>.
(gdb) x/30x $r0-$r12
0x10098 <_start+36>: 0xe3a07001 0xfef00000 0x000200ac
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x10     16
r3      0x10     16
r4      0x0      0
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10098 0x10098 <_start+36>
cpsr   0x10     16
(gdb)

```

We see the program exit after moving 1 to r7.

```

pi@raspberrypi:~ 
File Edit Tabs Help
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x1e     30
r3      0x10     16
r4      0x0      0
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10098 0x10098 <_start+36>
cpsr   0x10     16
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x1e     30
r3      0x10     16
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x10098 0x10098 <_start+36>
cpsr   0x10     16
(gdb) stepi
20    svc #0 @_exit syscall
(gdb) info registers
r0      0x0      0
r1      0x6      6
r2      0x1e     30
r3      0x10     16
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      0x7efff520 0x7efff520
lr      0x0      0
pc      0x1009c 0x1009c <_start+40>
cpsr   0x10     16
(gdb) stepi
[Inferior 1 (process 7286) exited normally]
(gdb) info registers
The program has no registers now.
(gdb)

```

gdb run show the version of the raspberry pi and the license of the software.

The screenshot shows a terminal window titled "pi@raspberrypi: ~". The window has a blue header bar with the title and a menu bar below it. The main area of the terminal displays the following command-line session:

```
pi@raspberrypi:~ $ ./arithmetic2
bash: ./: Is a directory
pi@raspberrypi:~ $ as -g -o arithmetic2.o arithmetic2.s
pi@raspberrypi:~ $ ld -o arithmetic2 arithmetic2.o
pi@raspberrypi:~ $ ./ arithmetic2
bash: ./: Is a directory
pi@raspberrypi:~ $ ld -o arithmetic2 arithmetic2.o
pi@raspberrypi:~ $ gdb run
GNU gdb (Raspbian 7.12-6) 7.12.0.20161007-git
Copyright (C) 2016 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-gnueabihf".
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"...
run: No such file or directory.
(gdb) █
```

Appendix:

GitHub

qubitsGSU / Project2

Code Issues Pull requests Projects Insights Settings

Arm assembly parallel project 2
Updated 3 minutes ago

To do	In progress	Done
Justin Choi Make the report Added by jychoi910	JP Sacha Assist in ARM Assembly Programming and report Added by jychoi910	Adam Henrie Upload video and Parallel Programming Tasks Added by jychoi910
	Titilayo Shonuyi ARM Assembly Programming Added by jychoi910	Shanza Siddiqi Create To do/In progress/Done list along with cards Added by jychoi910

Important Links:

- Slack: computerorganizespr19.slack.com
- GitHub: <https://github.com/qubitsGSU>
- YouTube: <https://www.youtube.com/watch?v=um1jBlF9TRI>