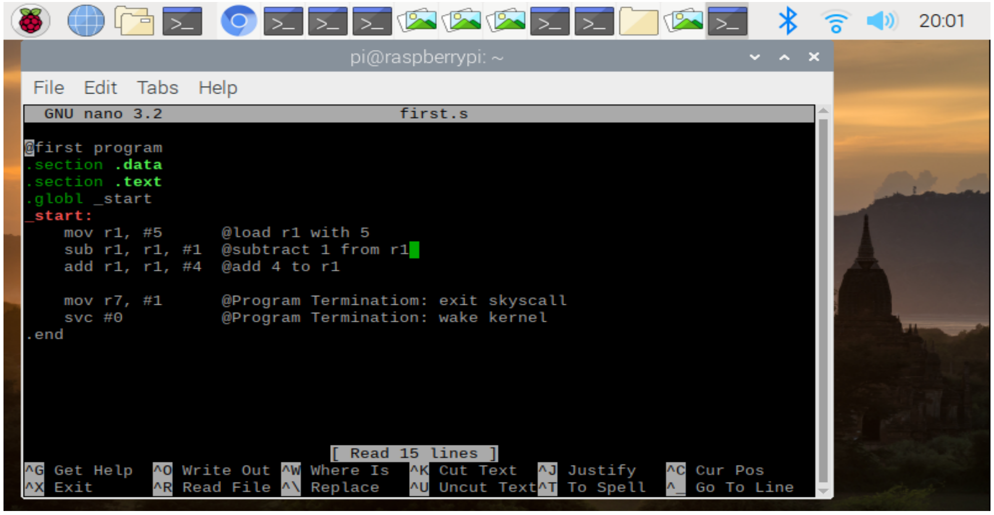
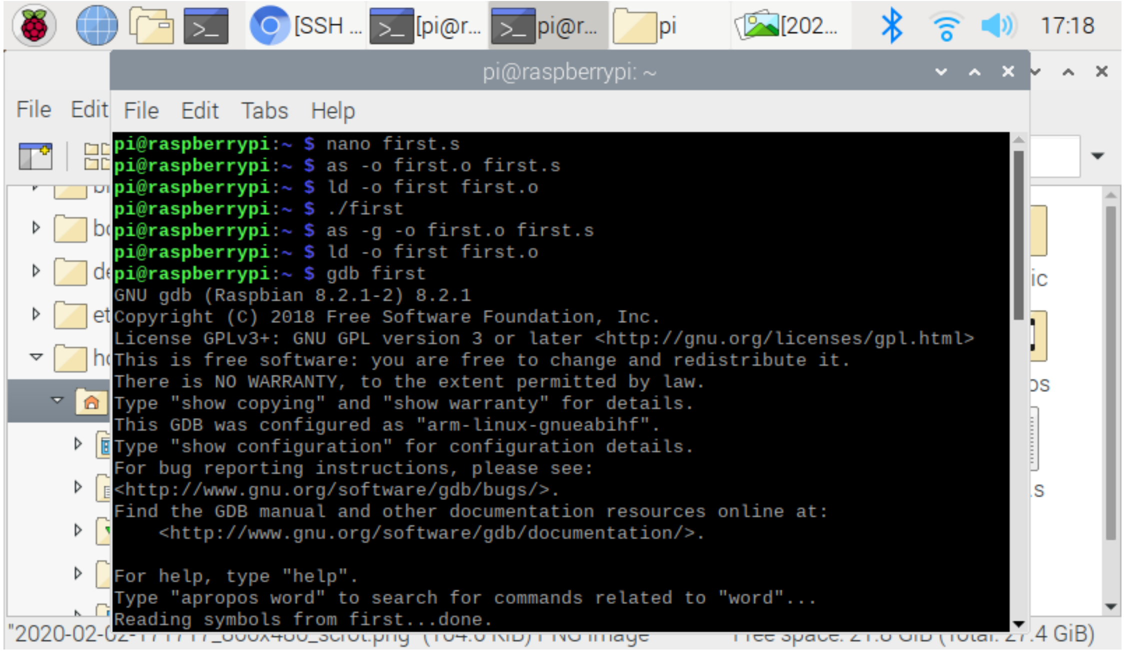


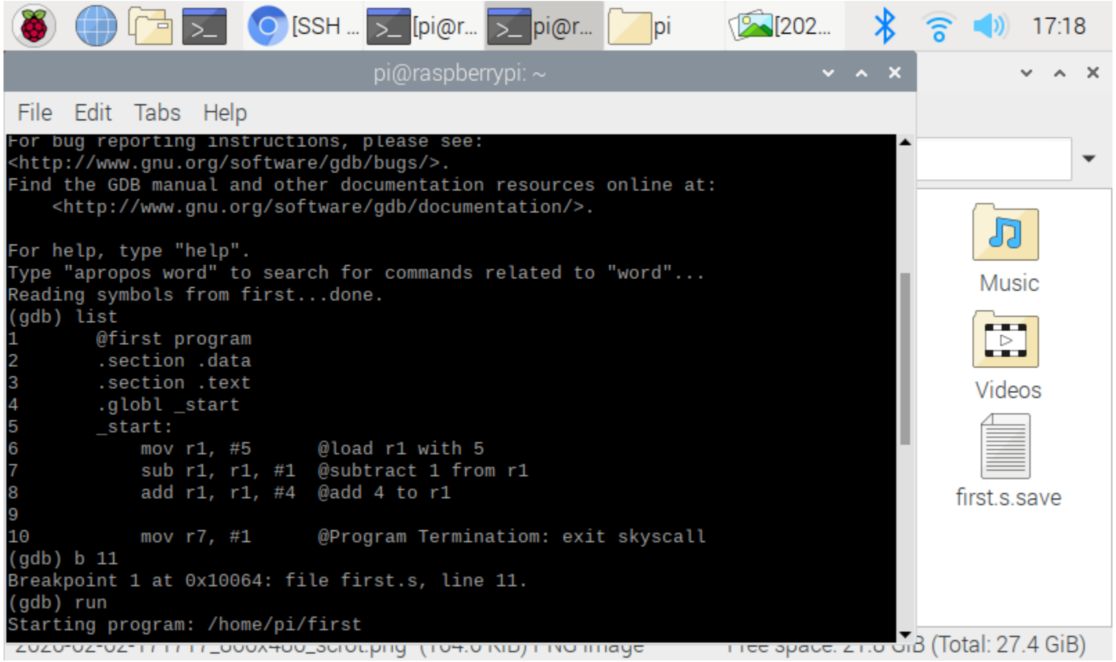
```
  GNU nano 3.2                         first.s

@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          @Program Terminatiom: exit skyscall
    svc #0              @Program Termination: wake kernel
.end

                                [ Read 15 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
```

This is the source code for the first.s file in nano editor.



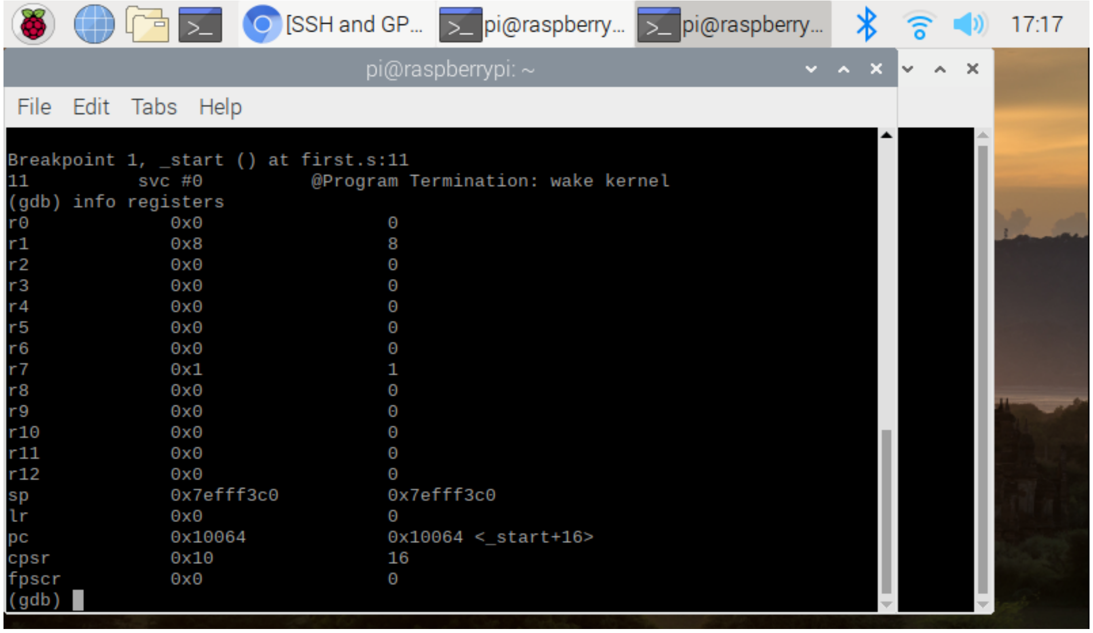
```
pi@raspberrypi:~ $ nano first.s
pi@raspberrypi:~ $ as -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ ./first
pi@raspberrypi:~ $ as -g -o first.o first.s
pi@raspberrypi:~ $ ld -o first first.o
pi@raspberrypi:~ $ gdb first
GNU gdb (Raspbian 8.2.1-2) 8.2.1
Copyright (C)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

This is assembling the first.s file, linking it to get an executable, and running it using the ./ first command.



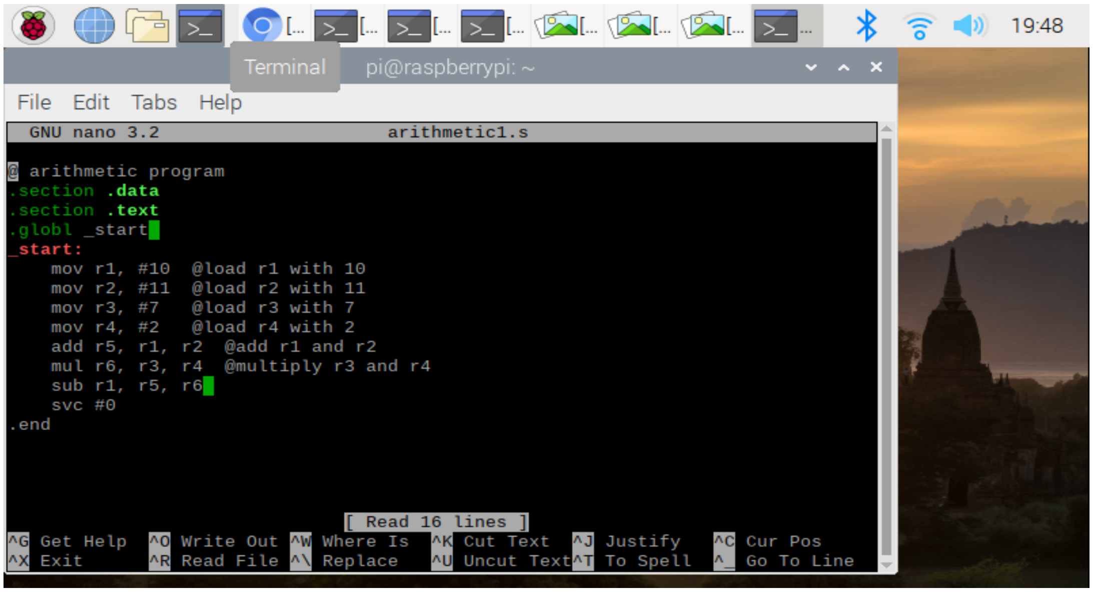
```
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

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 Terminatiom: exit skyscall
(gdb) b 11
Breakpoint 1 at 0x10064: file first.s, line 11.
(gdb) run
Starting program: /home/pi/first
```

This is going to launch the debugger to link machine code to source code. With the command gdb 11, the code stops executing before line 11.



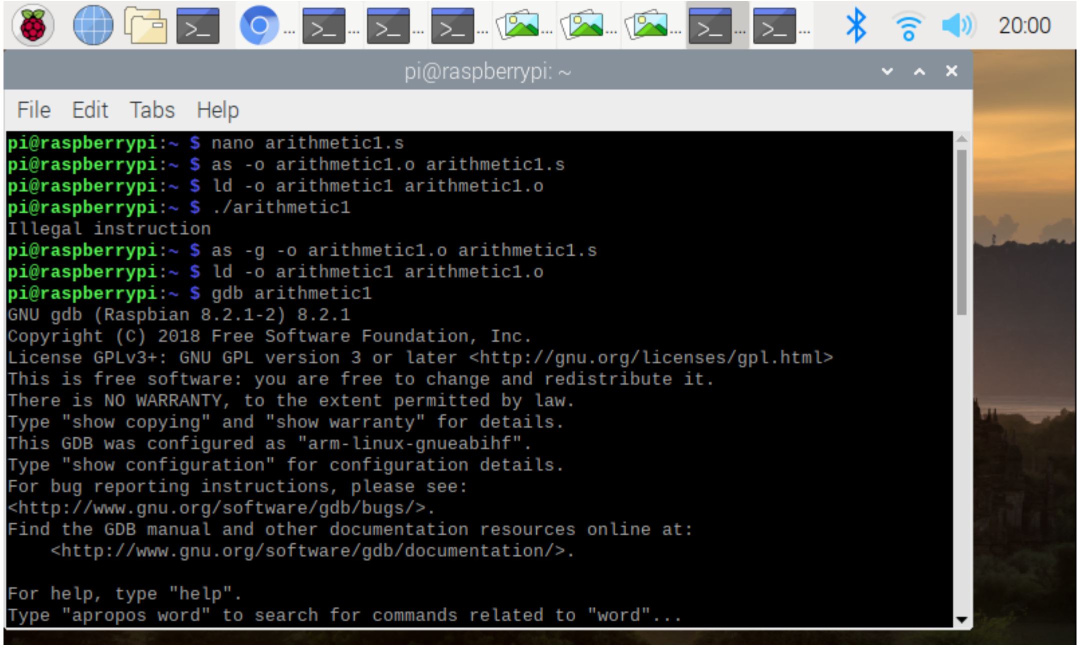
```

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

This examines the registers. All registers except r1 and r7 are empty. This is because the code loaded values (mov) only to r1 and r7.



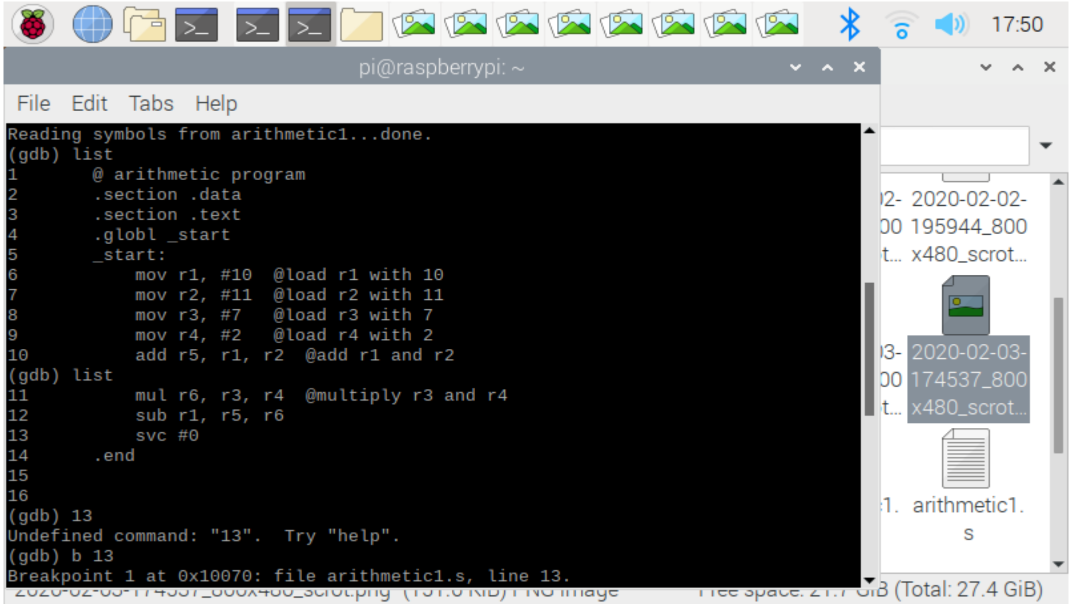
```
  GNU nano 3.2                     arithmetic1.s

@ arithmetic program
.section .data
.section .text
.globl _start
_start:
    mov r1, #10  @load r1 with 10
    mov r2, #11  @load r2 with 11
    mov r3, #7   @load r3 with 7
    mov r4, #2   @load r4 with 2
    add r5, r1, r2  @add r1 and r2
    mul r6, r3, r4  @multiply r3 and r4
    sub r1, r5, r6
    svc #0
.end

                                [ Read 16 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
```

This is the source code for the arithmetic1.s file in nano editor.



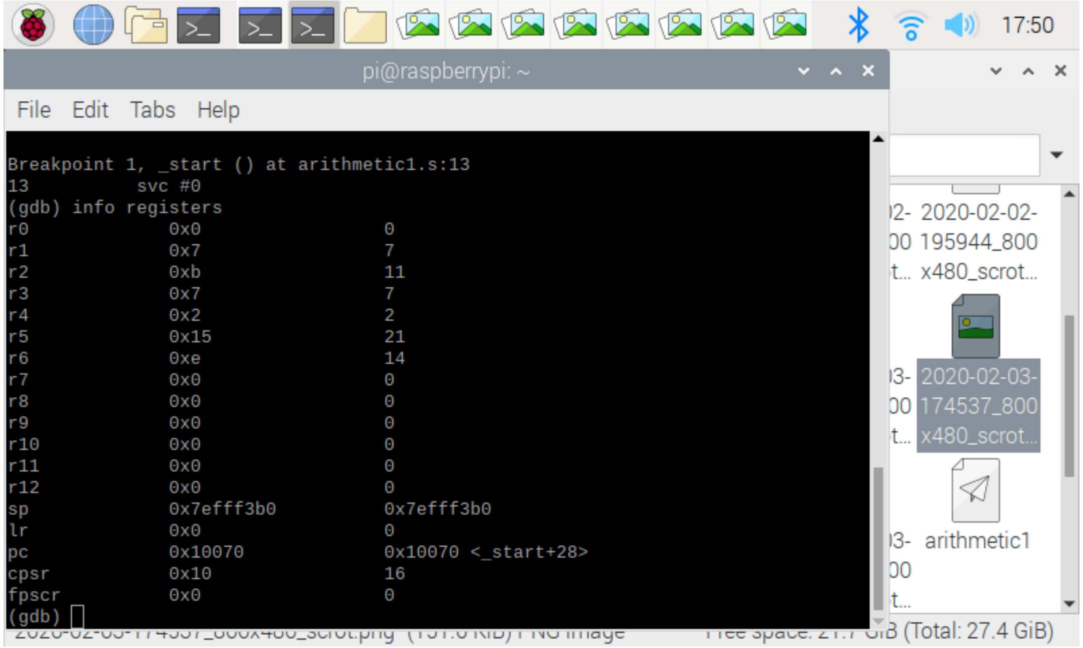
```
pi@raspberrypi:~ $ nano arithmetic1.s
pi@raspberrypi:~ $ as -o arithmetic1.o arithmetic1.s
pi@raspberrypi:~ $ ld -o arithmetic1 arithmetic1.o
pi@raspberrypi:~ $ ./arithmetic1
Illegal instruction
pi@raspberrypi:~ $ as -g -o arithmetic1.o arithmetic1.s
pi@raspberrypi:~ $ ld -o arithmetic1 arithmetic1.o
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-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"...
```

The arithmetic1.s file is being assembled, linked to get an executable, and run using the ./arithmetic1 function. It produces an “illegal instruction” error.



```
Reading symbols from arithmetic1...done.
(gdb) list
1       @ 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           mov r3, #7   @load r3 with 7
9           mov r4, #2   @load r4 with 2
10          add r5, r1, r2  @add r1 and r2
(gdb) list
11          mul r6, r3, r4  @multiply r3 and r4
12          sub r1, r5, r6
13          svc #0
14      .end
15
16
(gdb) 13
Undefined command: "13".  Try "help".
(gdb) b 13
Breakpoint 1 at 0x10070: file arithmetic1.s, line 13.
```

This is launching the debugger using the gdb arithmetic1 command. It set the breakpoint for line 13. It will allow the user to examine registers while the program runs.



```
Breakpoint 1, _start () at arithmetic1.s:13
13          svc #0
(gdb) info registers
r0             0x0                 0
r1             0x7                 7
r2             0xb                 11
r3             0x7                 7
r4             0x2                 2
r5             0x15                21
r6             0xe                 14
r7             0x0                 0
r8             0x0                 0
r9             0x0                 0
r10            0x0                 0
r11            0x0                 0
r12            0x0                 0
sp             0x7efff3b0          0x7efff3b0
lr             0x0                 0
pc             0x10070             0x10070 <_start+28>
cpsr           0x10                16
fpscr          0x0                 0
(gdb) 
```

This examines the registers while the program is running. R2-R4 are loaded with values (11,7,2) B, C, and D. It will perform the arithmetic operation (A+B)-(C*D) which results in the final R1 value being 7. R5 contains the sum of A,B (11,10) which is 21. R6 contains the product of C,D (7,2) which is 14.