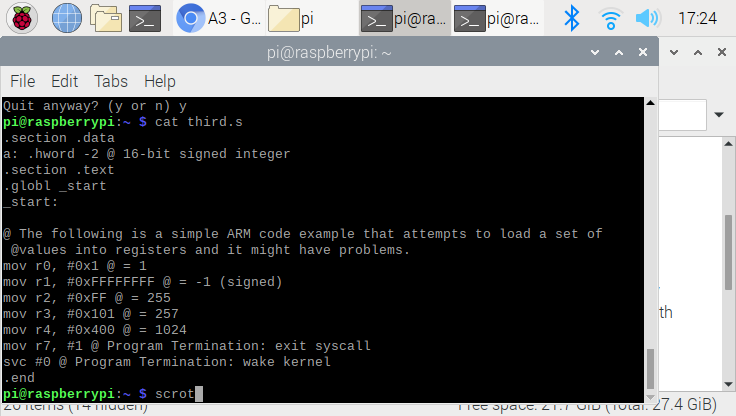
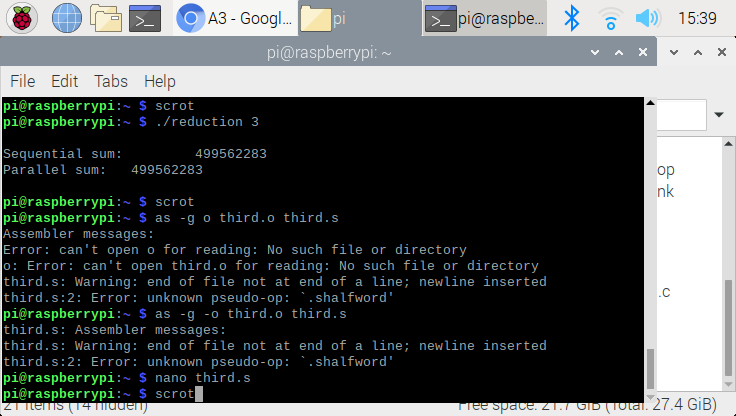
Modified third code



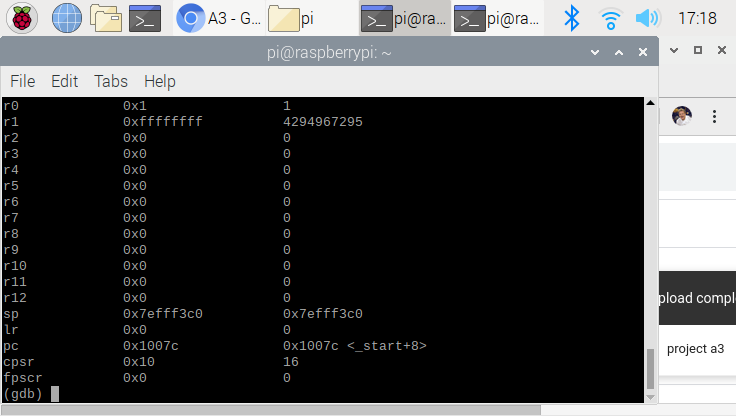
The code has been modified from .shalfword to .hword. Without the modification, the code will not compile since it has the error(invalid data type). This code is directly from the pdf instruction.

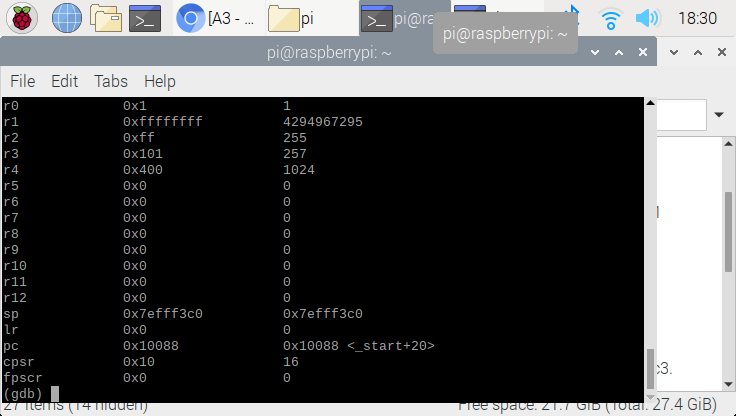
Error message



This is the error message if we declare the “.shalfword”. The compiler would not recognize the command, and throw the error. In order to correct the problem, we have to do with hword instead.

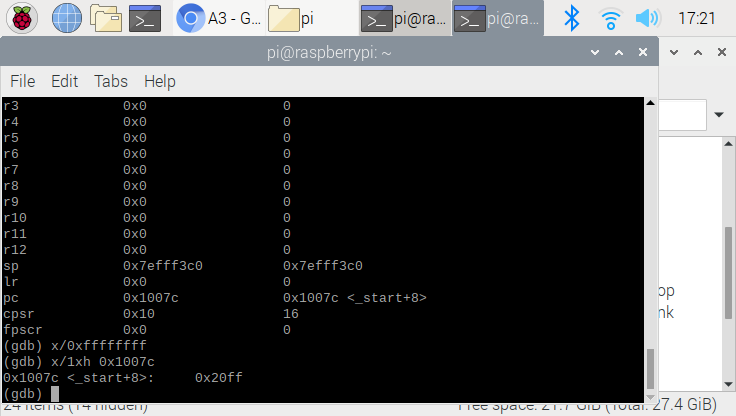
Register Info





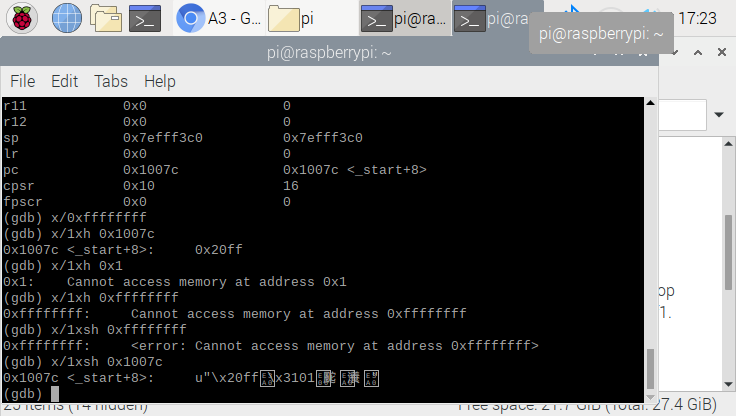
The first picture is the register info when break point is located at line 7. Second picture is the register info when the break point is located at the end of the file. In this exercise, we used the “stepi” command to go through the line by line to observe the changes of its register value.

Access memory with h



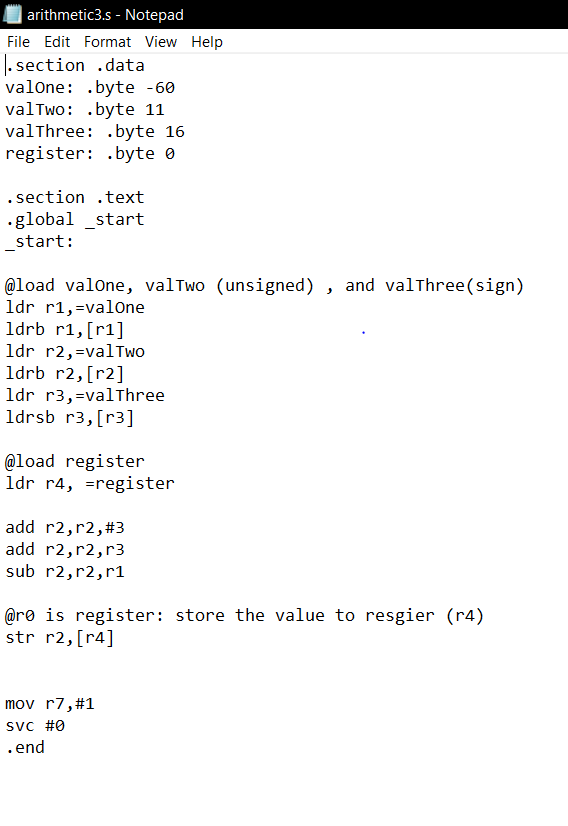
Looking at the last line, I used address ofo “0x1007c. When using h, the value is given in the hexadecimal form (0x20ff).

Access memory with sh



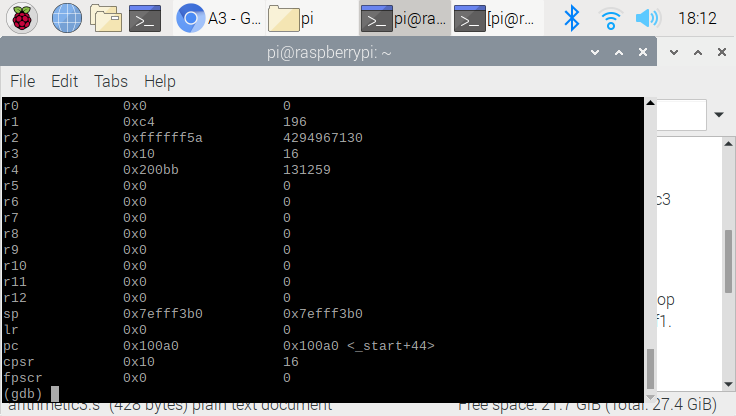
Look at the very last line. It is exactly same code but have sh instead of h. The result of the command was not readable since the text is broken. This tells that the type is unmatched.

**Arithmetric3**



This is the code for the Arithmetics3. In this program, we performed “register= val2 +3 + val3 -val1”. The program is to understand how the negative number work. The inststrctuion requires us to declare the ValOne, and ValTwo as unsigned, and valThree as the signed number. We also learned how to declare the signed and unsigned number in .code (which is ldrb and ldrsb).

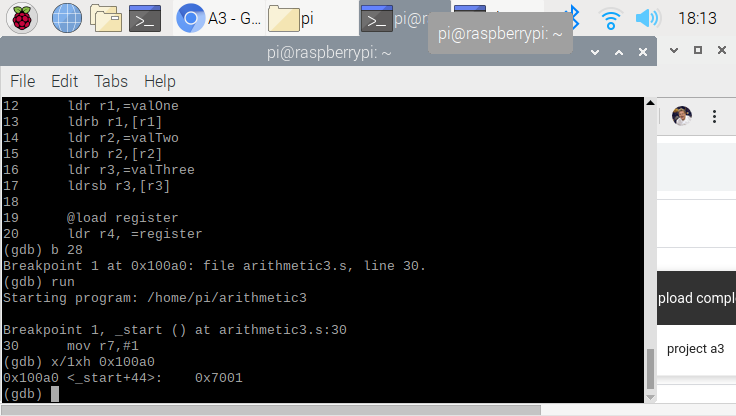
Register info



This is result of the memory and resgier

One unique thing about this register is that the valOne(which is -60) is stored as 196. Since the computer can’t distinguish the positive and negative, the number is stored in as the 2’s compliment. Also the result (register) is stored in r4. It is quite interesting to see that the result of our math is different from the computer prespective. (131259 != -30)

Access memory pratice



This is another practice for the memory access.