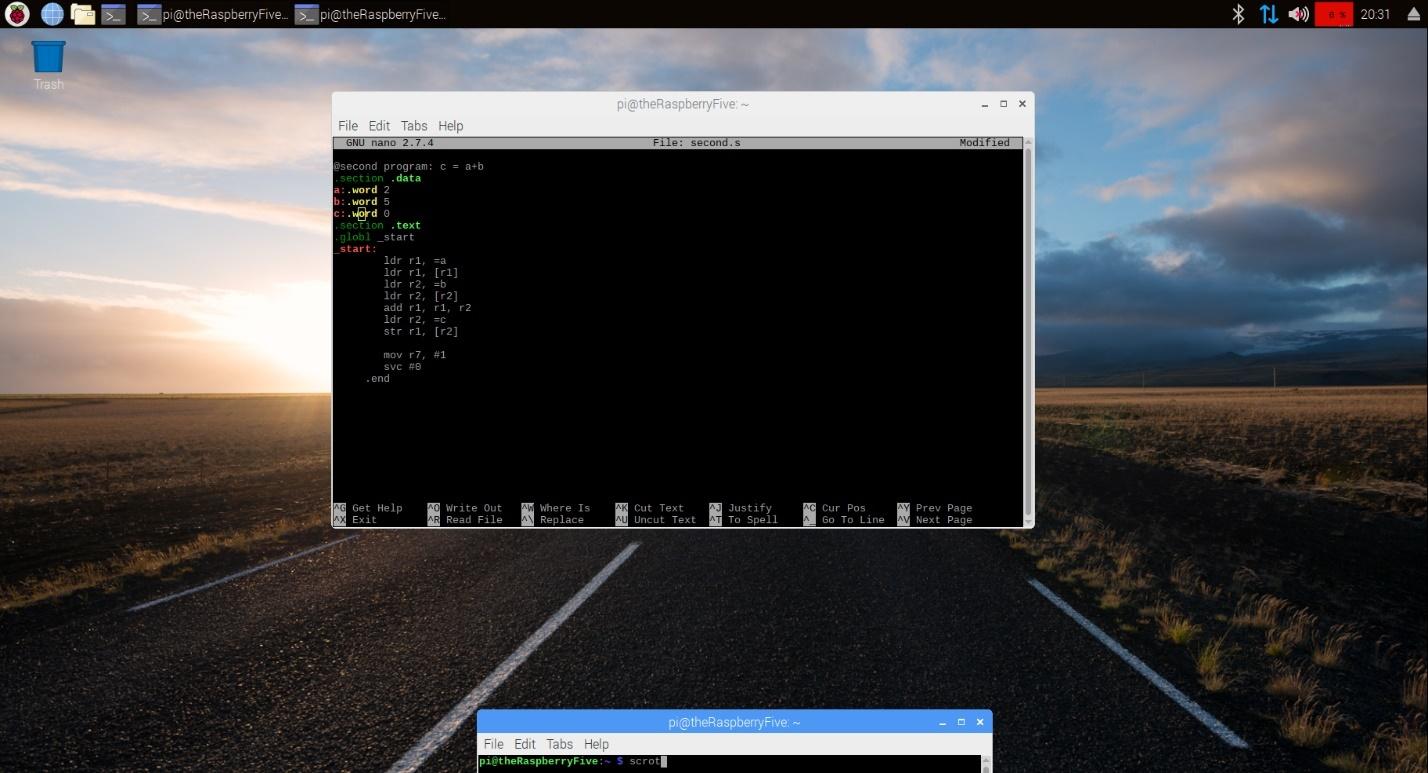
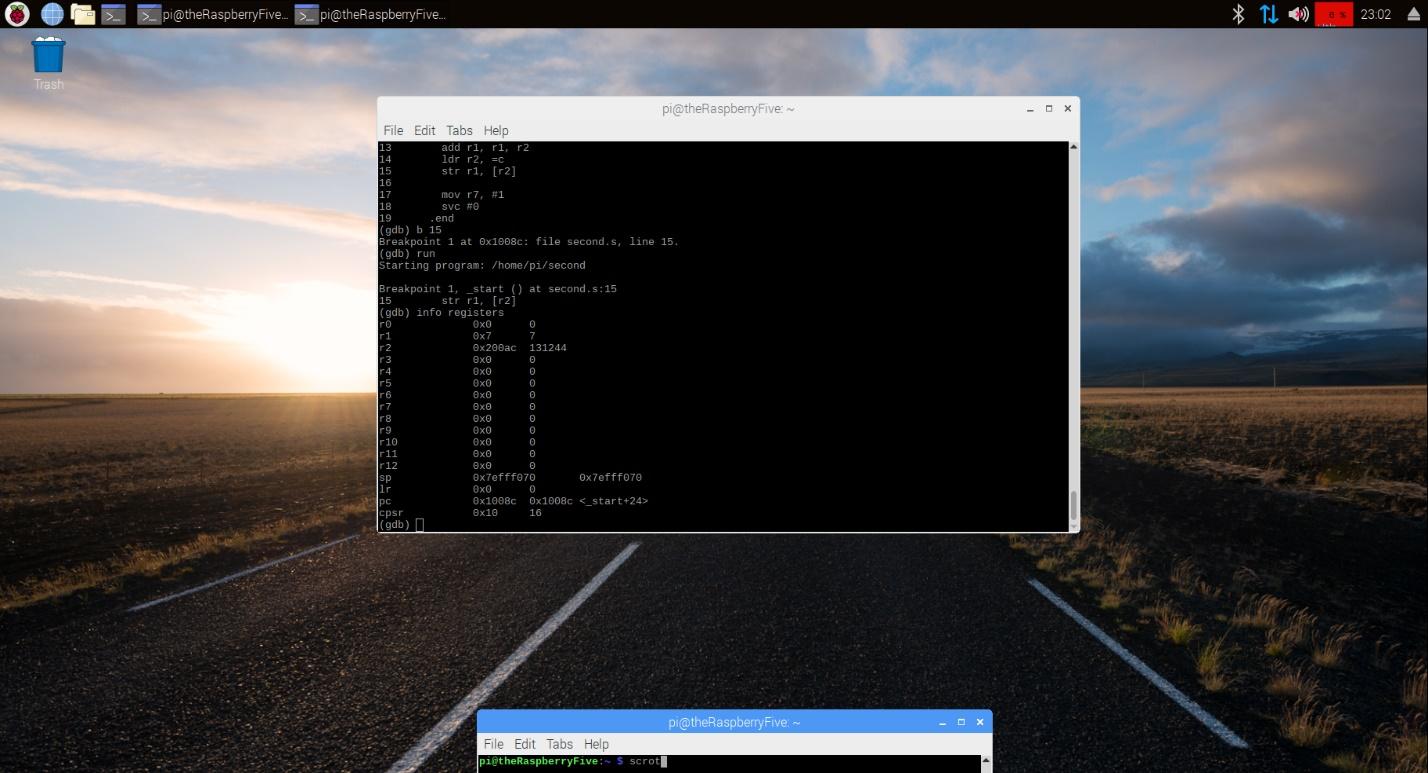
I, Jacques Agbenu, oversaw the ARM programming. The instructions were followed with some difficulty, but the outcome was the one that was sought after.

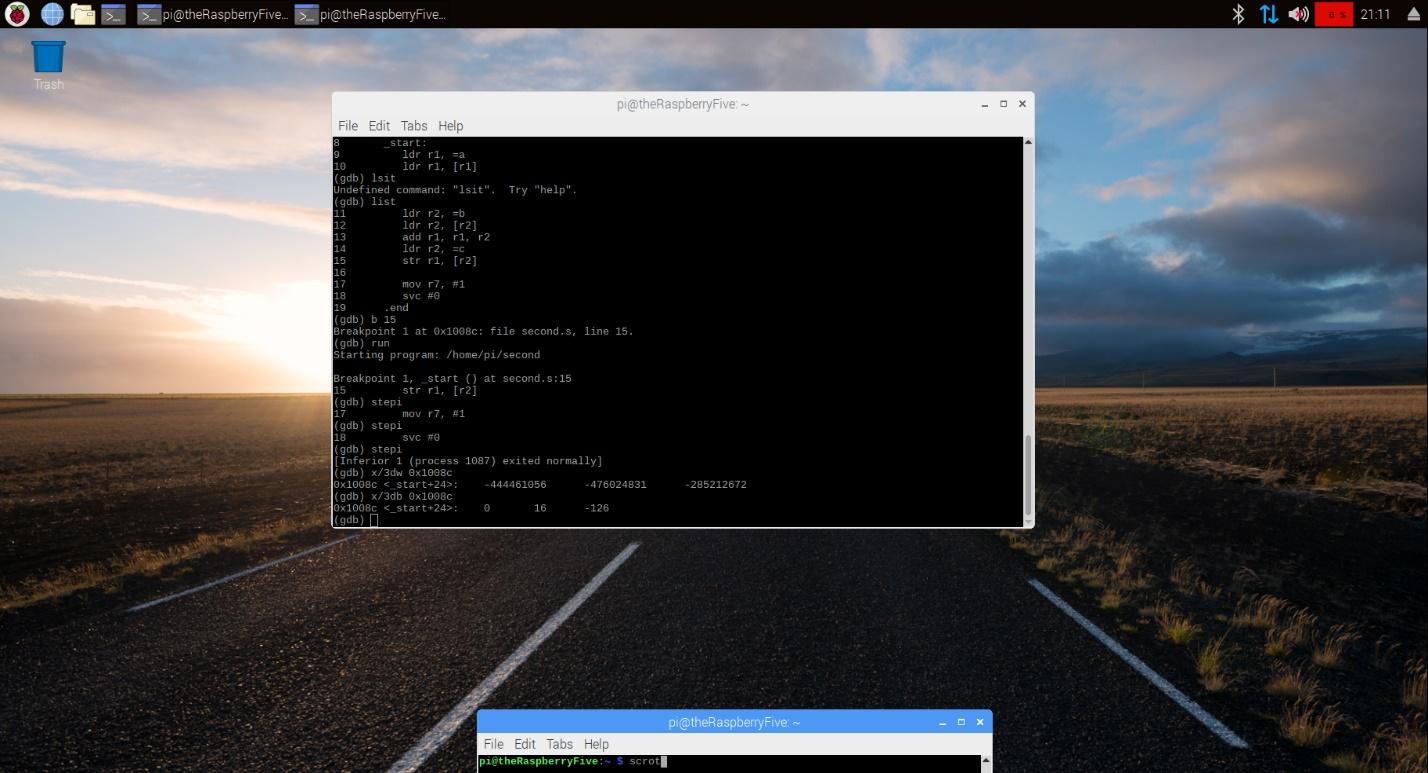
The code for part 1 of the ARM programming was a bit different than the first ARM programming that was done. This part had a new aspect that wasn’t seen in the programming before. The loading of the address to the register and then loading the value of that address to the same register was a bit hard to wrap my head around but after a few minutes of looking at the code, I was able to understand what was happening.



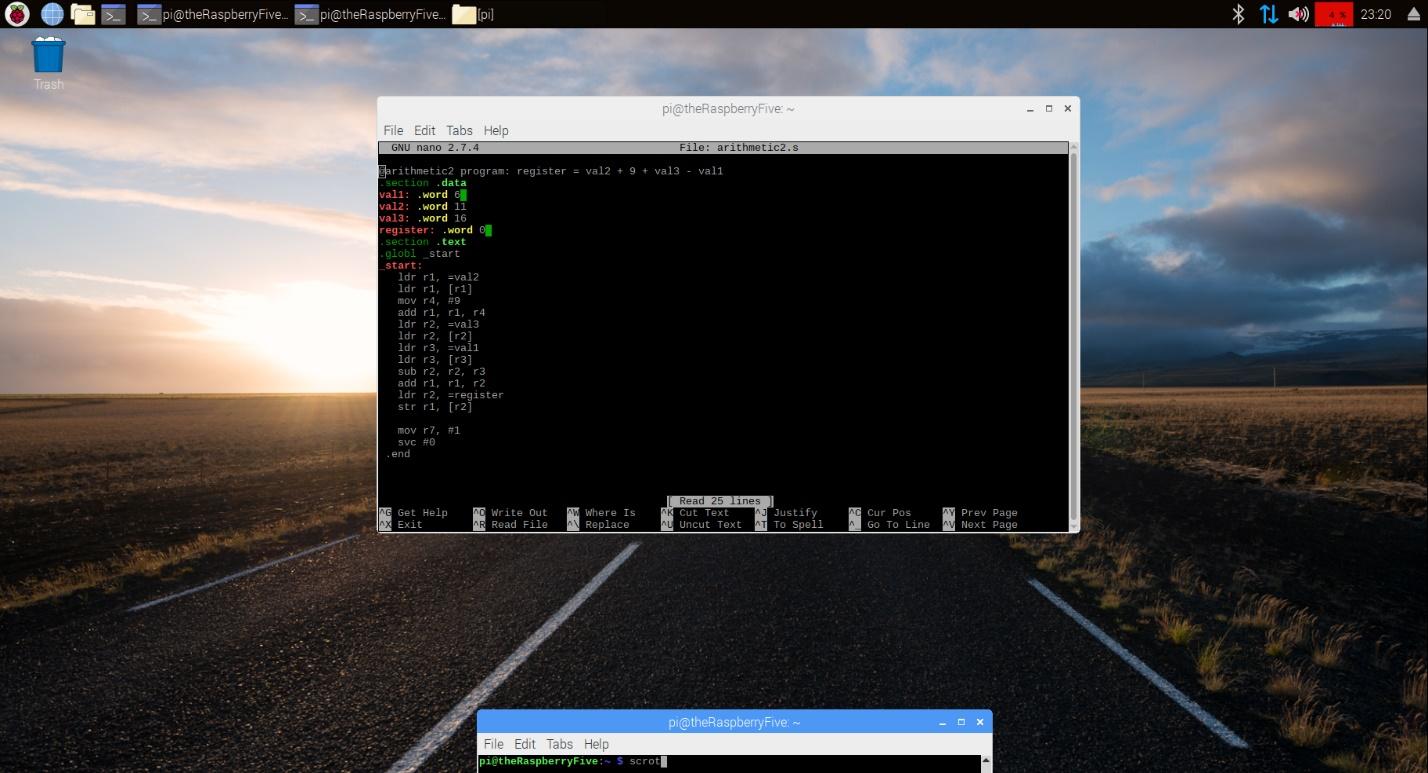
The register output of this code came out like it was supposed to, although it might not have been necessary to include this part into the report, but I decided to add it anyway.



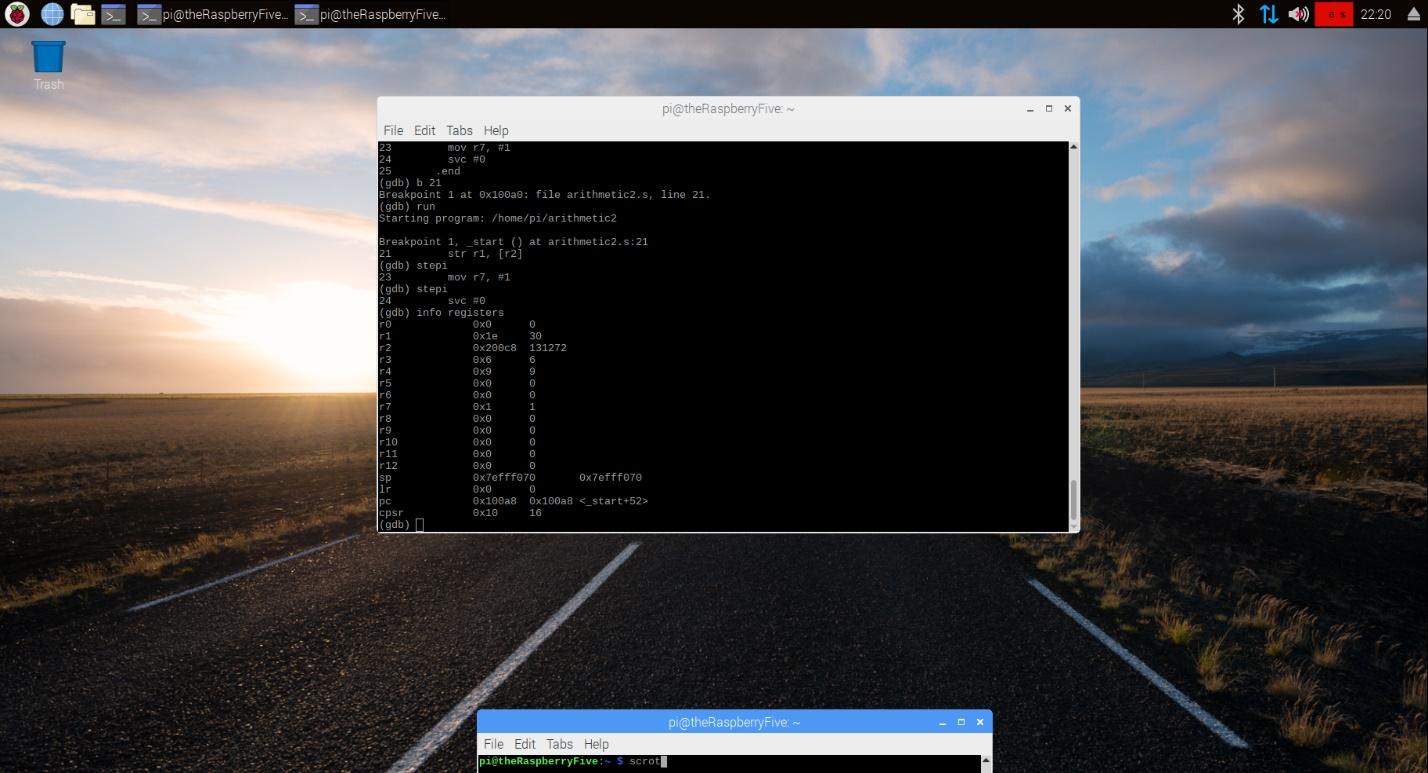
Looking up the three words was a confusing task. We have put breakpoints in ARM code before, but we have never used that command to show that type of result before.



Part two of the ARM programming was a bit harder than the first part and required a bit more time than the first time. The loading of the variables to registers were easy but moving an immediate to the register was confusing. I thought that I would have to load the immediate to a register like the other values, but it seems I didn’t have to do that. The use of registers after they are empty was a very useful feature that let me keep my code concise and clean.



Finding the value of this snippet of code was the same as before. The GBD provided the info registers and that was where the value was found , in decimal and in hexadecimal.



The last output that was needed was the 3 words in hexadecimal of this snippet of code was easy to come by as the same was done with the first snippet of code.