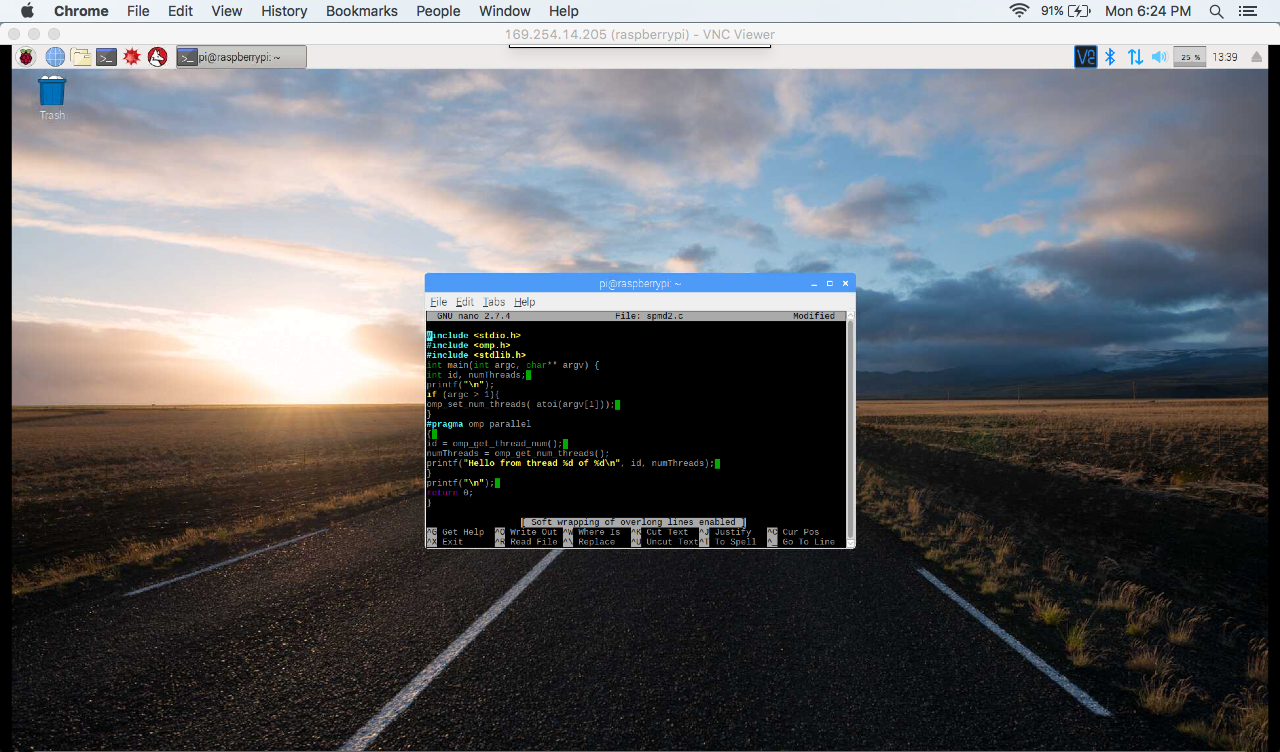
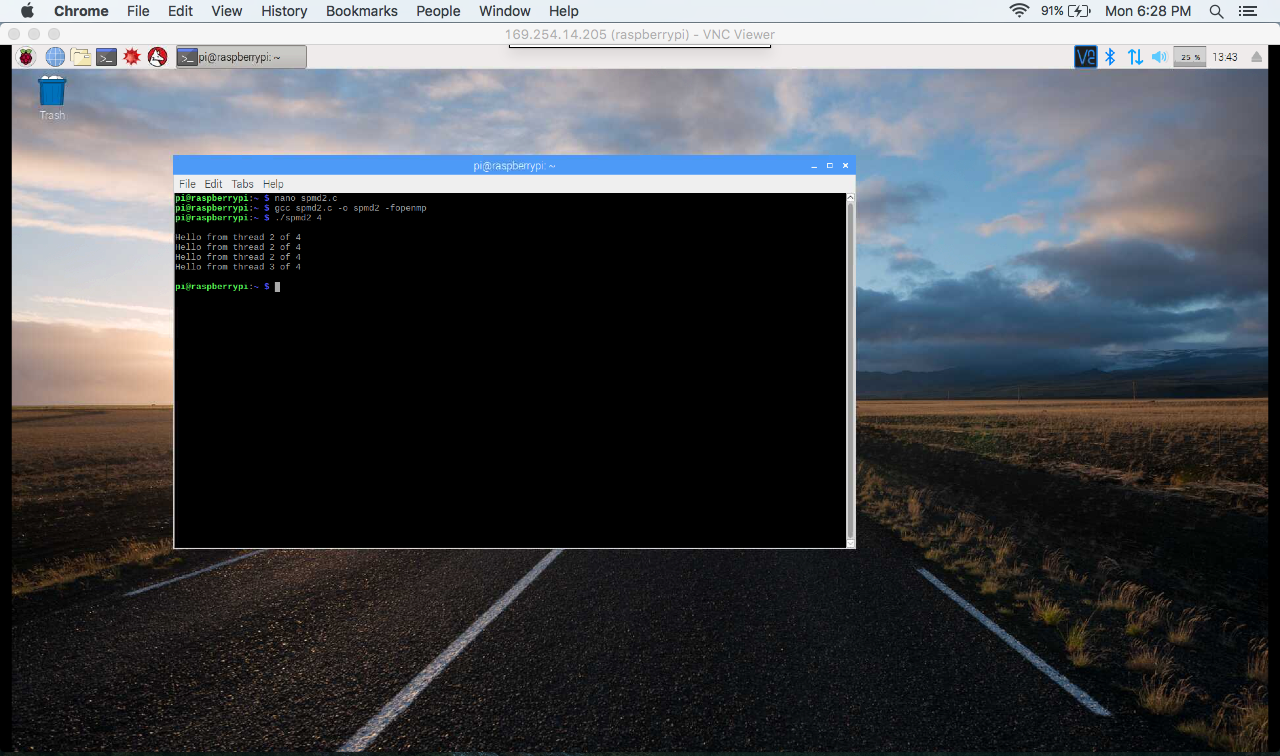
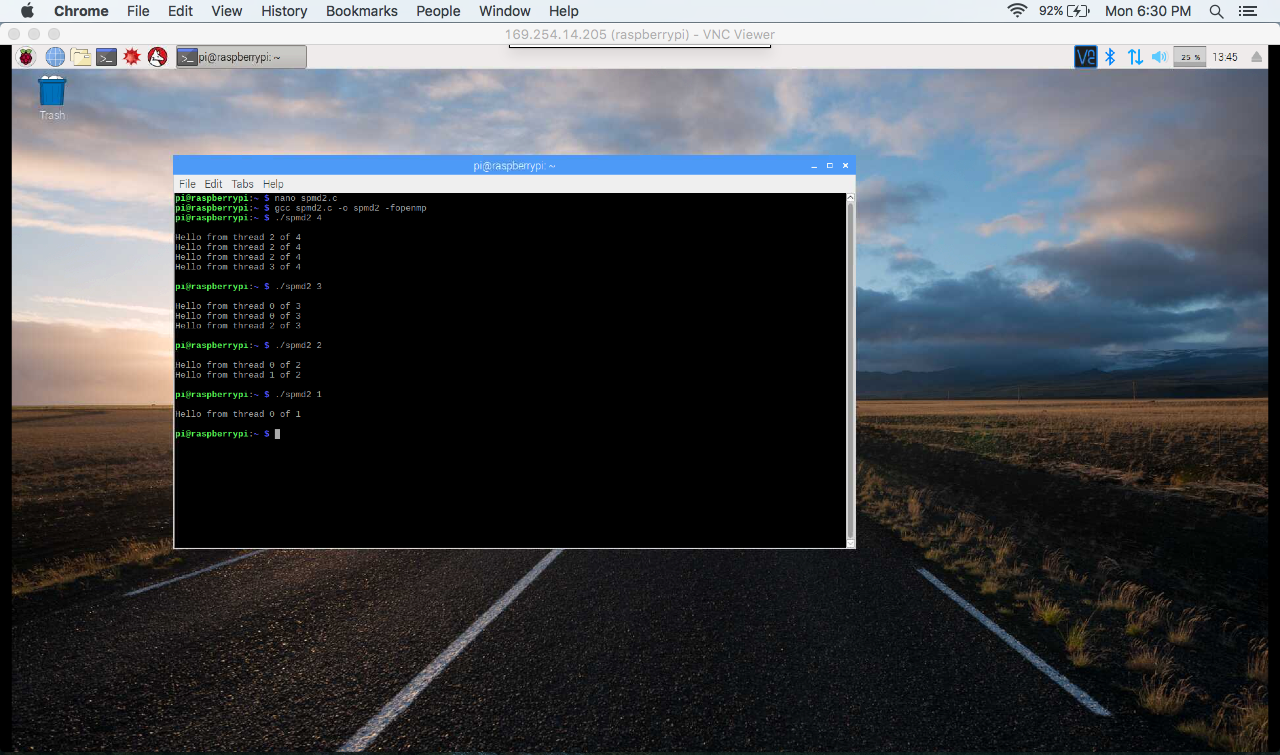
**Parallel Programming Basics**



* In this screenshot, the initial code given from the spmd2.c program is inputted.



* In this screenshot, 4 threads were forked from the command line-argument ./spmd2 4
* What I found interesting was seeing the thread id number ‘2’ show up more than once for 4 threads.



* In this screenshot, I tried to run the program again, but this time with 3, 2, and 1 threads being forked.
* What was interesting to me was when 4 threads were forked, the thread id number ‘2’ appeared more than once (3 times), but when 3, 2, or 1 threads were forked, the thread id number ‘0’ appeared for all of them.
* Each thread is supposed to be given its own unique id, so the thread id number can’t appear more than once.
* Each thread is supposed to keep track of its id separately, which is why it cannot have the same one in different threads.



* In this screenshot, the spmd2.c program is edited to where declaring the variables is changed so that a thread id number does not appear more than once.
* Declaring ‘id’ and ‘numThreads’ as int fixes the thread id number issue and now each thread will have their own private copy of the variables names ‘id’ and ‘numThreads’.



* In this screenshot, the new code for the spmd2.c program is executed and produces the following output when for 4, 5, 6, and 7 threads are forked.
* Since each thread has their own private copy of the variables ‘id’ and ‘numThreads’, thread id numbers do not appear more than once.
* Each thread has their own unique id and is keeping track of its id separately.
* The thread id number is no larger than the number of threads declared in the command line argument.