The first thing I did was look at make\_data.c. The only things I did were parsing the command line arguments and adding error checking. The file looks like this:

A screenshot of a computer program

Description automatically generated

Next, I looked at read\_data.c. All I did there was parse the command line arguments. The file looks like this:

A screen shot of a computer program

Description automatically generated

The utilities.c file is where most of the work happened. I included the stdlib.h as well as time.h for this assignment. This is so I can use malloc() rand(), srand() and time(). The first function, allocate\_memory() is simple. It looks like this:

A computer screen shot of text

Description automatically generated

The second function, initialize\_memory() is also simple. It looks like this:

A computer screen with text and numbers

Description automatically generated

This initializes each index of the array to a random int.

The write\_memory\_to\_file() function again was pretty simple. It looks like this:

A computer screen with text and images

Description automatically generated

All this does is write the contents of the array into the file.

The determine\_file\_size() function looks like this:

A screenshot of a computer program

Description automatically generated

This function was pretty simple as well. The trickiest part was figuring out how to actually get the file size. I looked it up and I saw that SEEK\_END automatically finds the end of the file, which made the rest of the function simple from there.

The initialize\_memory\_from\_file() function was the hardest one for me. It looks like this:

A screenshot of a computer program

Description automatically generated

For some reason, it was adding a bunch of extra zeros at the end and I don’t know why. I just kept messing around with this function until it worked. I think it had something to do with line 120, but that was one of the first things that I changed and it was still adding the zeros. (unless I never saved it)

The final function, print\_memory() was simple as well. It looks like this:

A computer screen with white text

Description automatically generated

All this does is loop through the array and print the elements.

Folder before everything was compiled:

A screenshot of a computer screen

Description automatically generated

Making:

A screen shot of a computer code

Description automatically generated

Testing usage statements:

A screen shot of a computer code

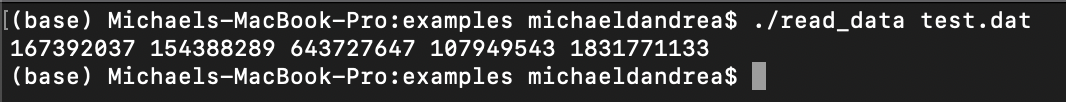
Description automatically generated

Making the file with 5 integers:

A screenshot of a computer

Description automatically generated

Reading the file:



Proving it is random:

A screen shot of a computer

Description automatically generated