Prog # 1

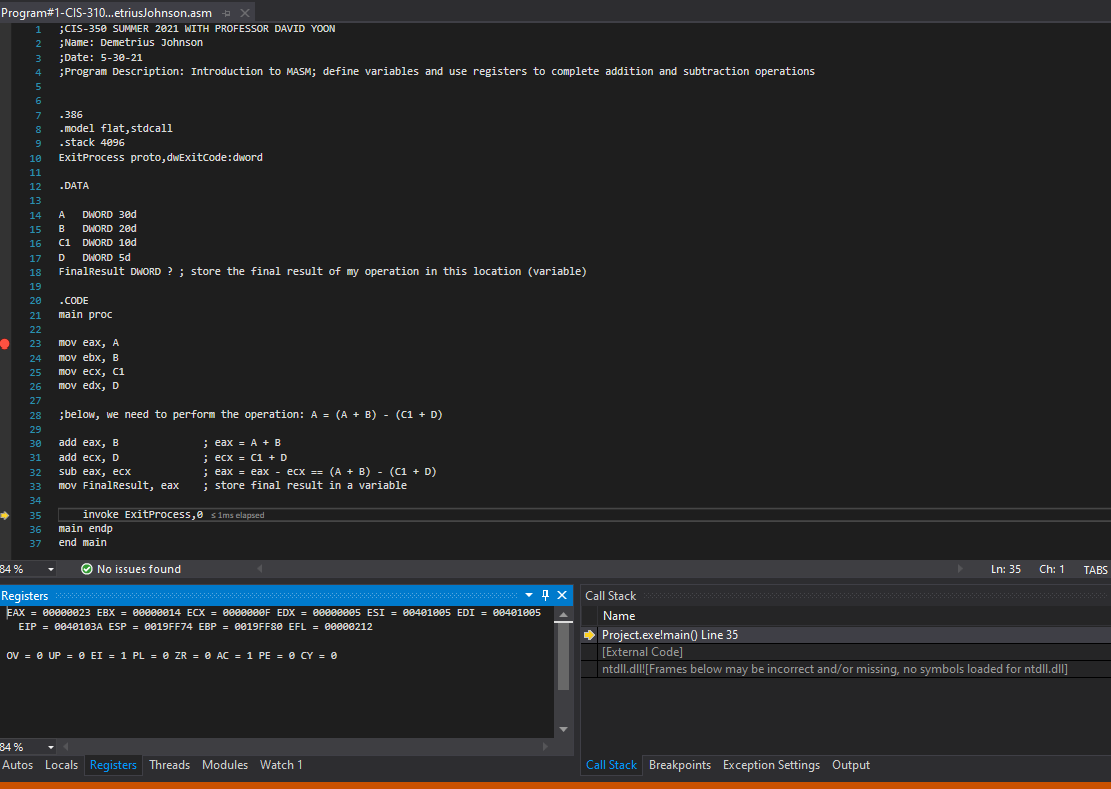
Demetrius Johnson

**CIS 310 (Yoon)**

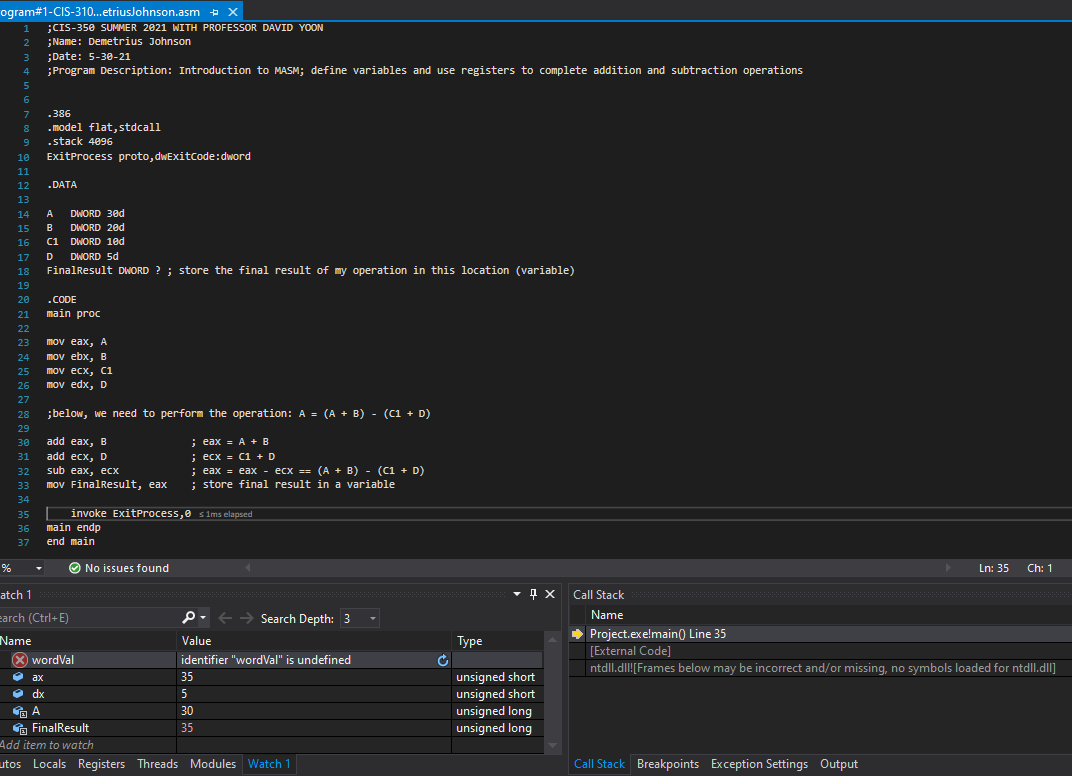
**June 2, 2021**

1. **Build/run the program without debugging.**
2. **Run the program in the debugging mode and add the register window to a debugging session (see Fig. 3.5).**

Notice, I added another variable to store the result of the arithmetic operation, called **FinalResult**; also in this screenshot I show what the value of all of my registers are after all operations have been completed:

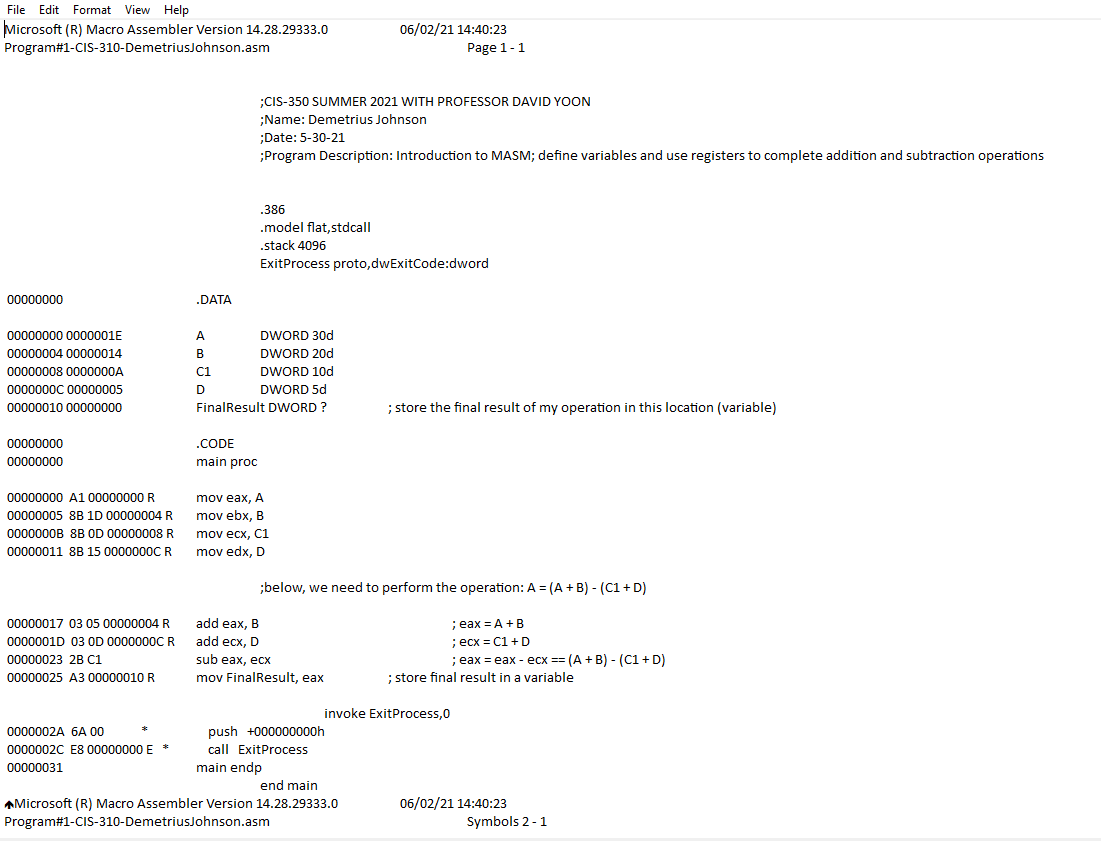


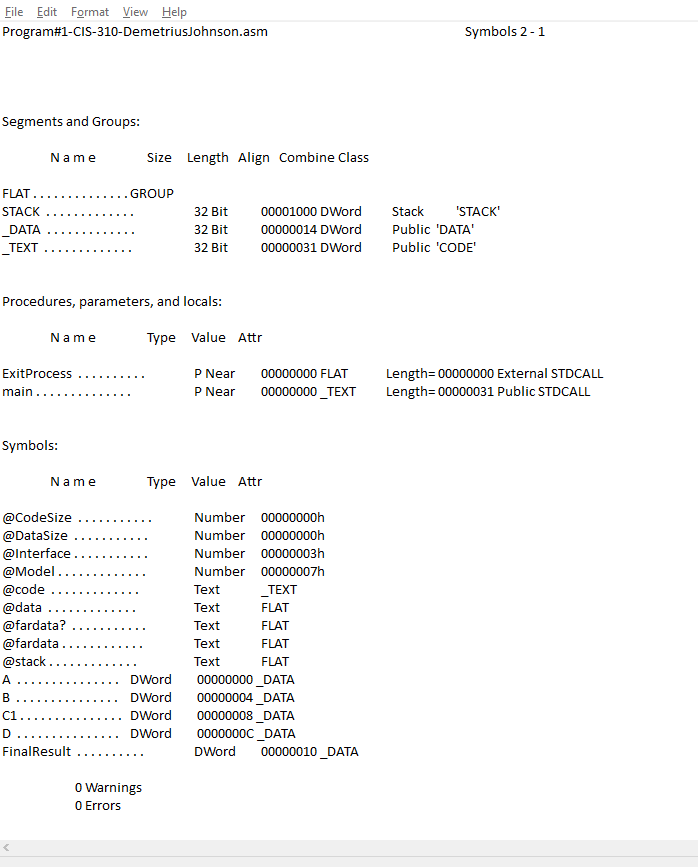
In this screenshot, I demonstrate what is stored in my variables and registers:



1. **Generate the listing file like the one in Fig. 3.8.**

In the next two screenshots, you see the listing file generated from my program. I read from the Irvine book that the memory locations correspond to some starting address relative to wherever the operating system begins assigning memory; as an example, the start memory location 00000000 (32-bit address represented as 8 hex digits) is really some other starting location value in memory, and thus every value after the start is also relative to that location. They simply generate the listing file starting from a relative 0-address location so it is easier for us programmers to identify and analyze memory usage and allocation that occurs for and during the program:





**Upload the screenshots from 1, 2 and 3.**