Make files use the same GNU that C programs use so they are almost interchangeable and all GNU documentation of make files can be applied.

To make a file be able to be debugged the make file must include the -g flag
Use the make command with ‘make DEBUG=1’ *If you use the one given where DEBUG = 1 is the if statement

Then enter the program by doing gdb “programname” if you gdb without the -g flag the program will open in the gdb but you will not be able to step through the program. You will also not be able to see any of the source code.

You can set a breakpoint from here with b “label name” This label name can be any of the labels inside of your assembly program. Most of the time you would want to start your debugger at the beginning of you program so you could do this by typing b _start, as long as you label the start of your program as _start

Then run the program using run or r. Once you have your break point you can use r to start the program. If there is no breakpoint set the program will just run normally.

You can step through the program with s. Stepping through the program can be very useful so you can see everything your program is doing. This is useful to look into registers to ensure the proper values you wanted to use are being used correctly and also allows you to see if and where your programing is jumping to.

You can read registers with ‘i r’. This will list every register with the values inside of them.

Other references
https://makefiletutorial.com/
https://www.tutorialspoint.com/gnu_debugger/gdb_quick_guide.htm