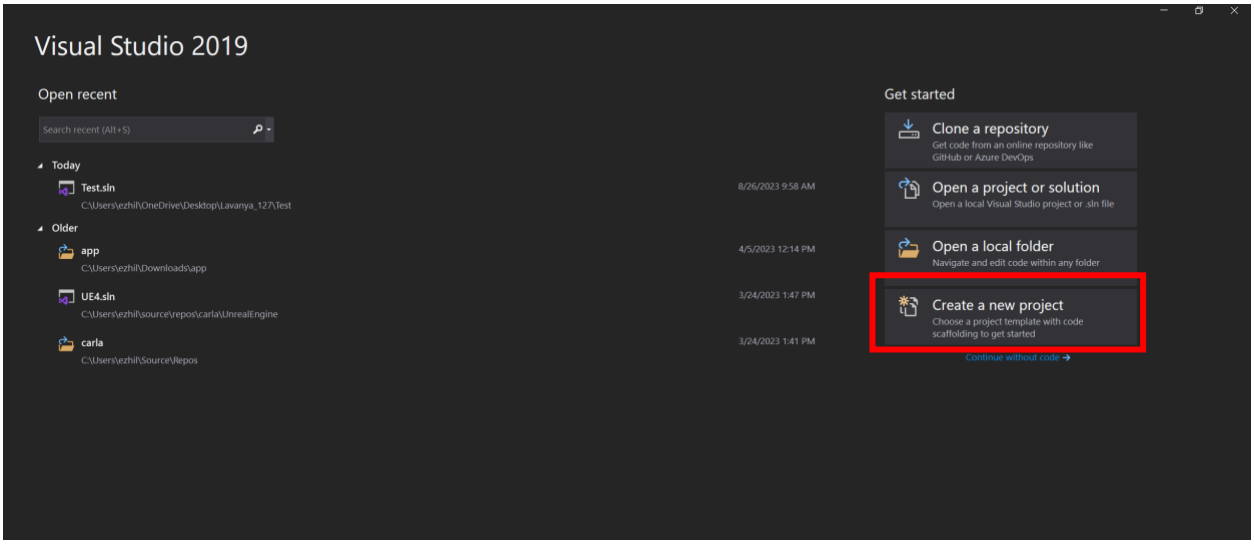
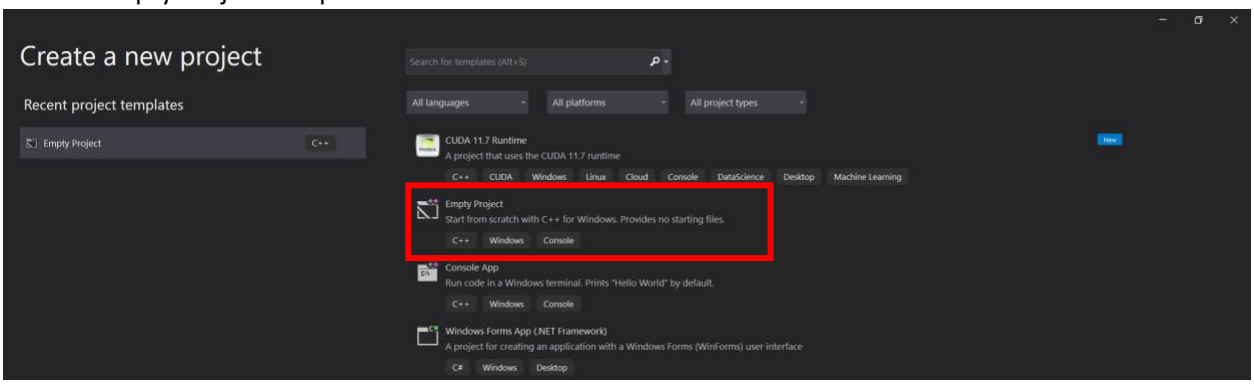
Steps to create a new project and debug

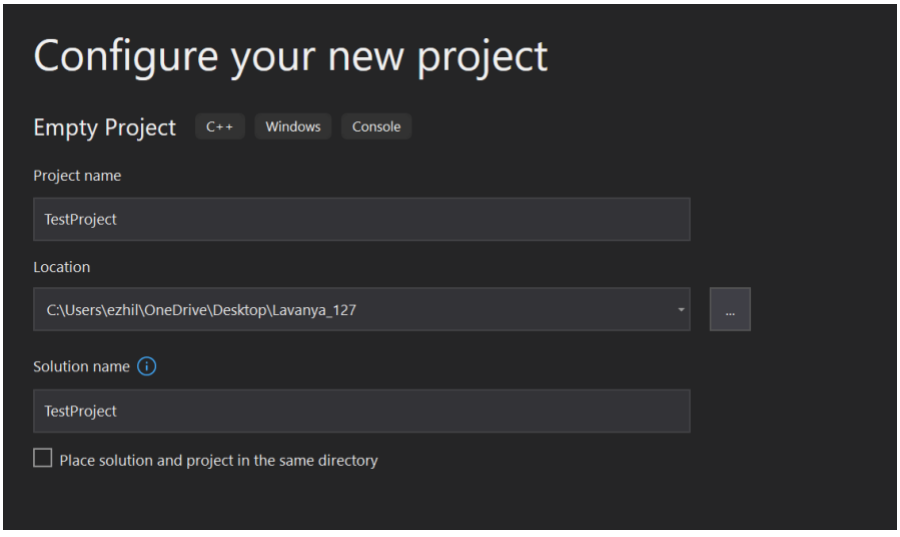
1. Open Visual Studio installed according to the steps mentioned in the Lab0 document.
2. Click on the create a new project:



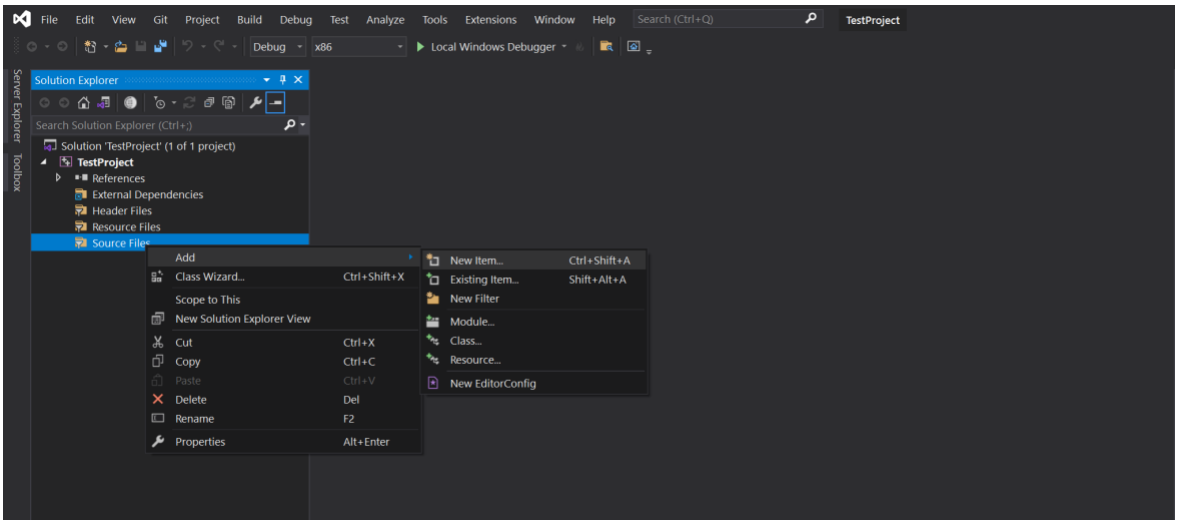
3. Choose Empty Project template and click next:



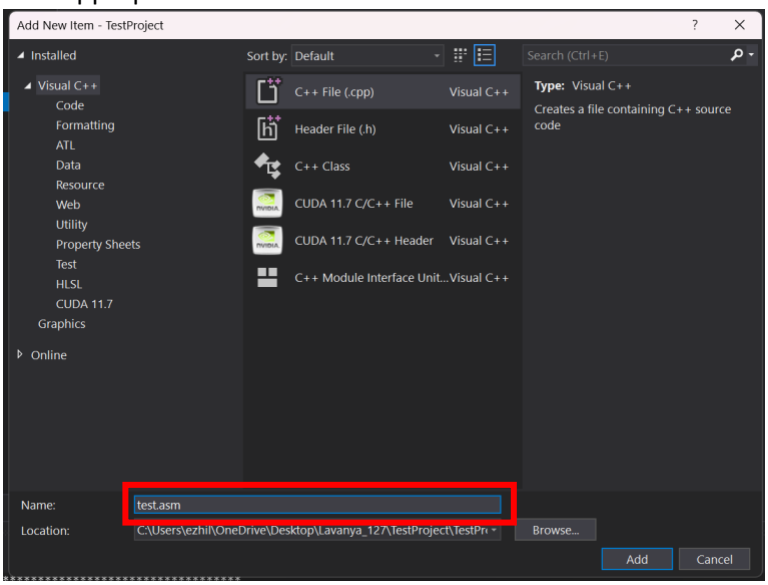
4. Give appropriate Project name and Location and click create:



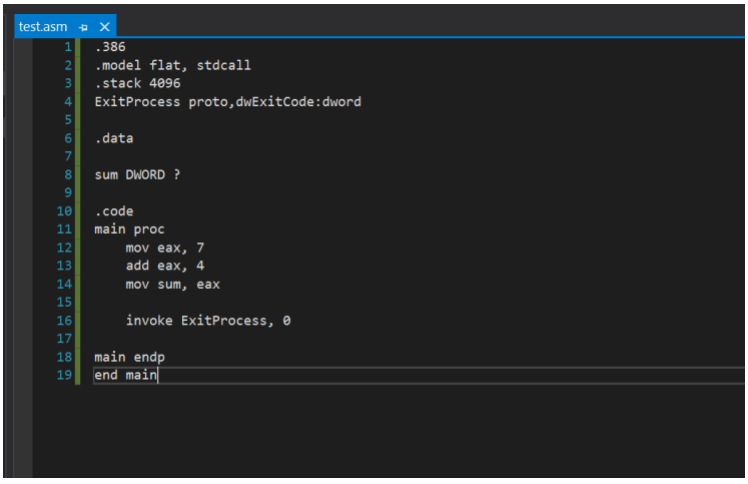
5. Your new project opens in the VS. Create a new source file by right-clicking on the source Files folder -> Add -> New Item



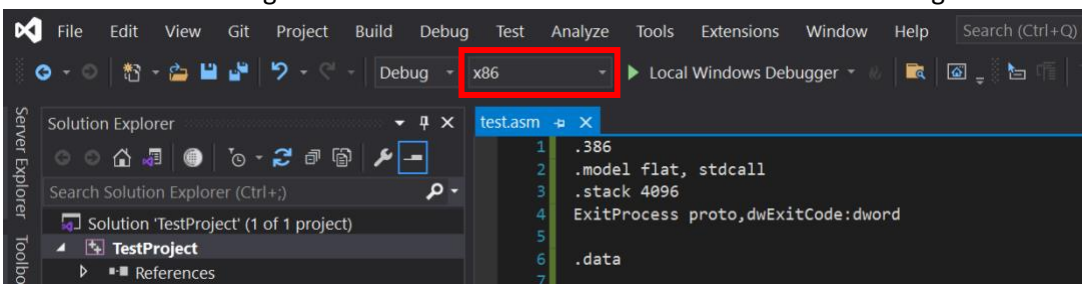
6. Give appropriate name for the source file and click on add:



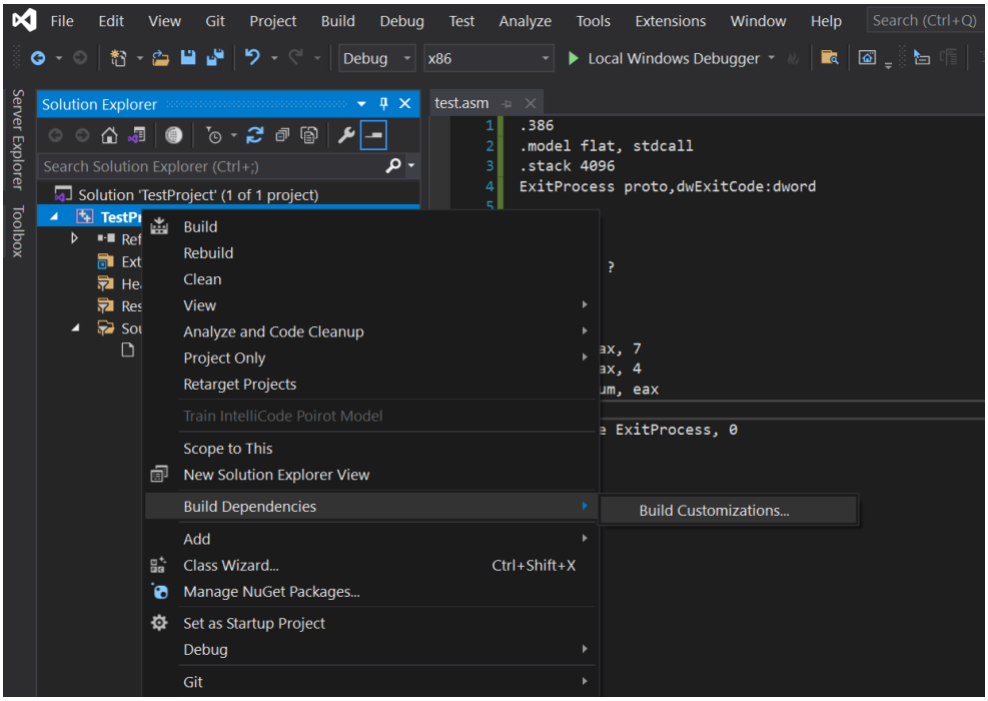
7. Now open the source file you just created and add your instructions and save the contents:



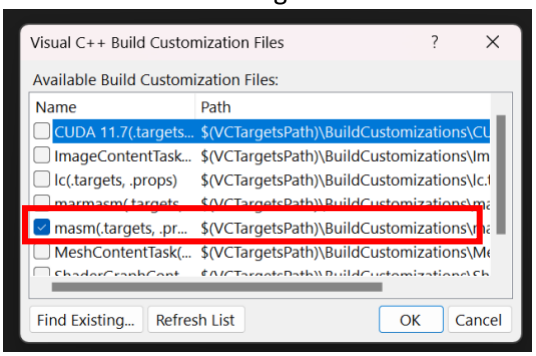
8. Now we need to configure it for MASM. In the menu bar choose x86 as the target architecture:



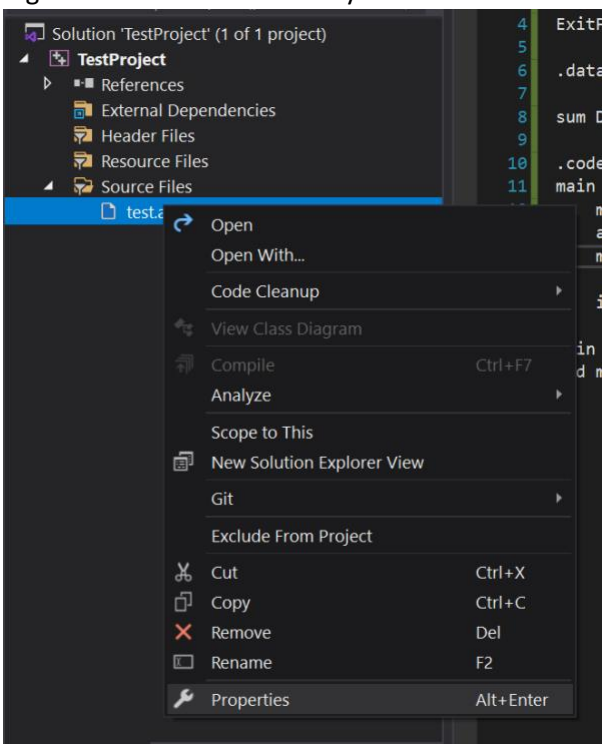
9. Right-click on the project -> Build dependencies -> Build customizations



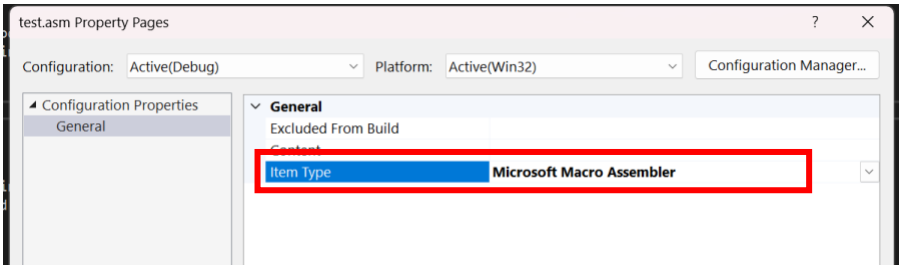
10. Enable the masm configuration file and click ok:



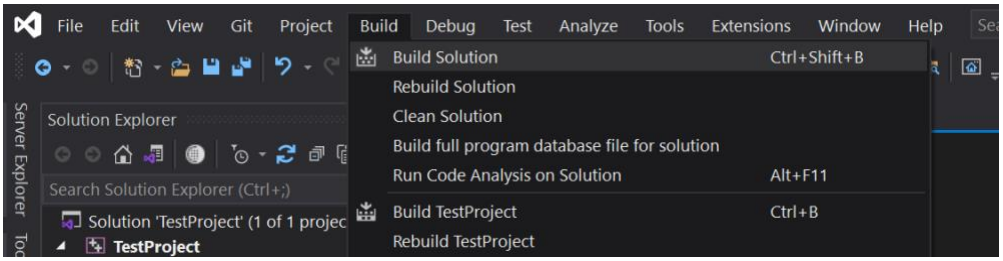
11. Right-click on the source file you created and click on properties:



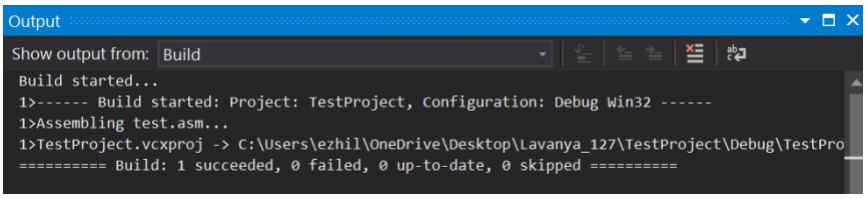
12. Choose Microsoft Macro Assembler from the dropdown for the ItemType and then click Apply and then OK:



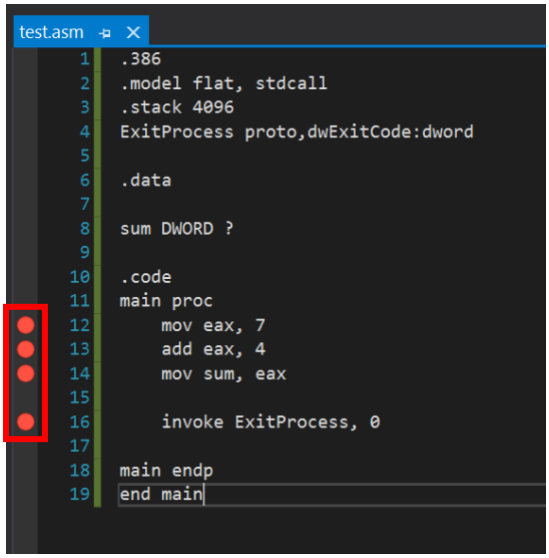
13. Now you are all set for the project build. To build the project click on Build -> Build Solution.



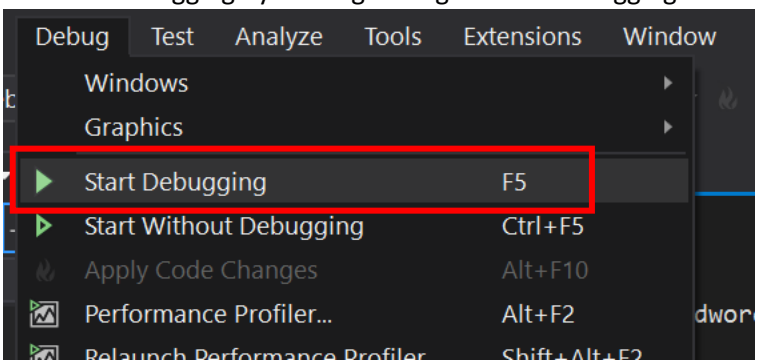
14. You should see a build successful message in the output window as shown below. If your code contains any errors the build fails showing the errors. You need to correct them and rebuild the solution.



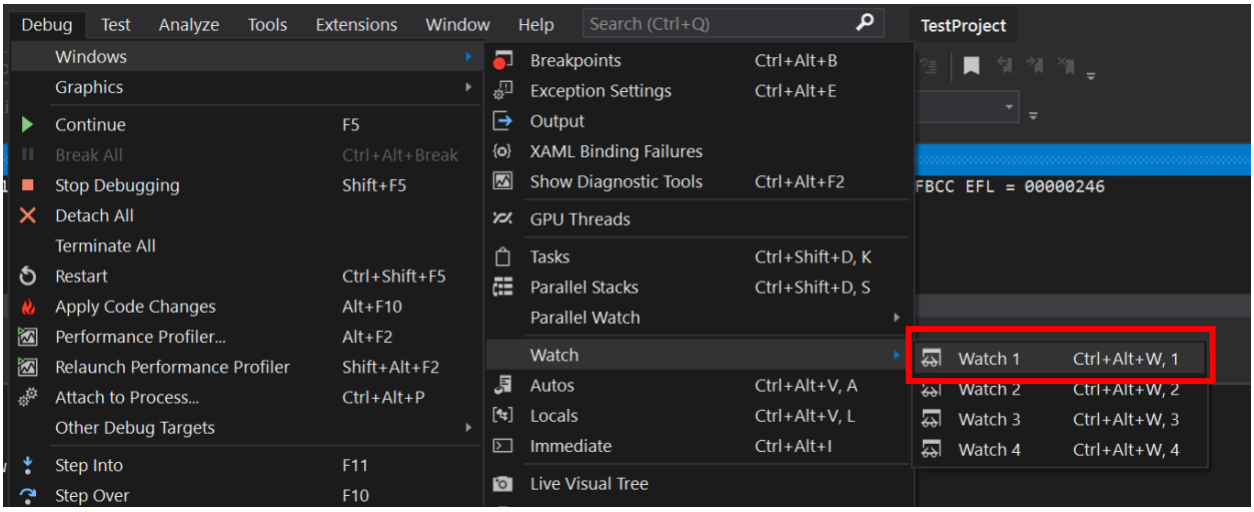
15. In order to debug the code you need to add breakpoints by clicking on the pane to the left of the source code:

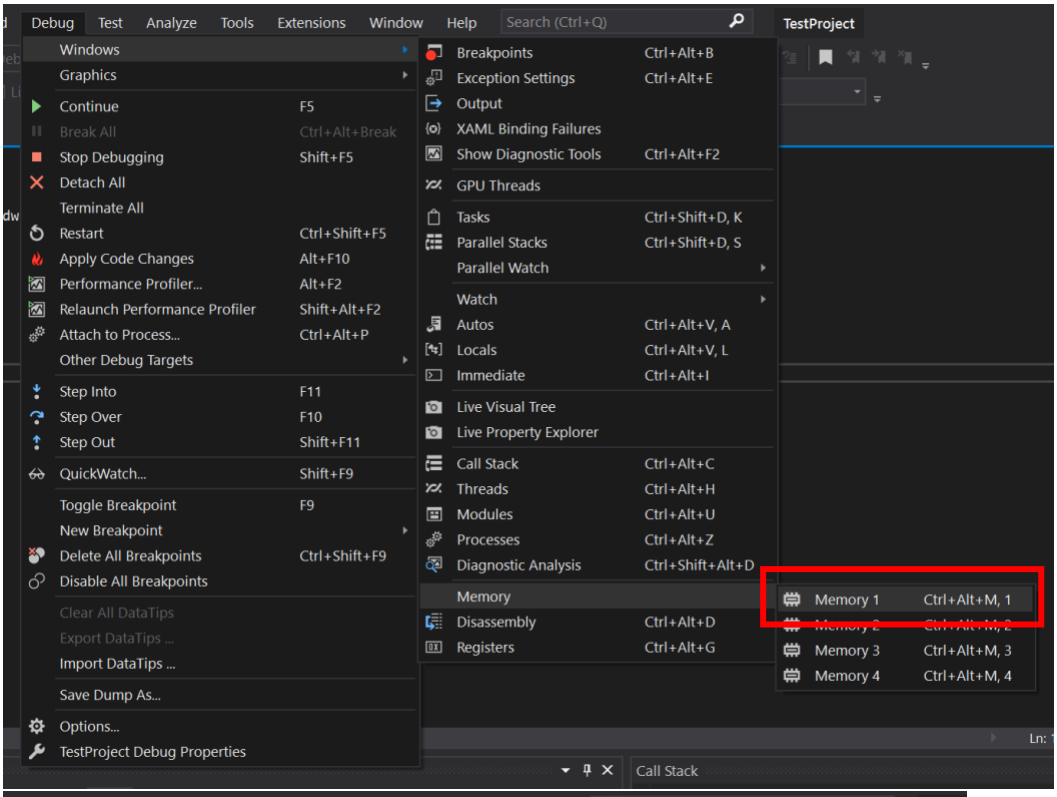


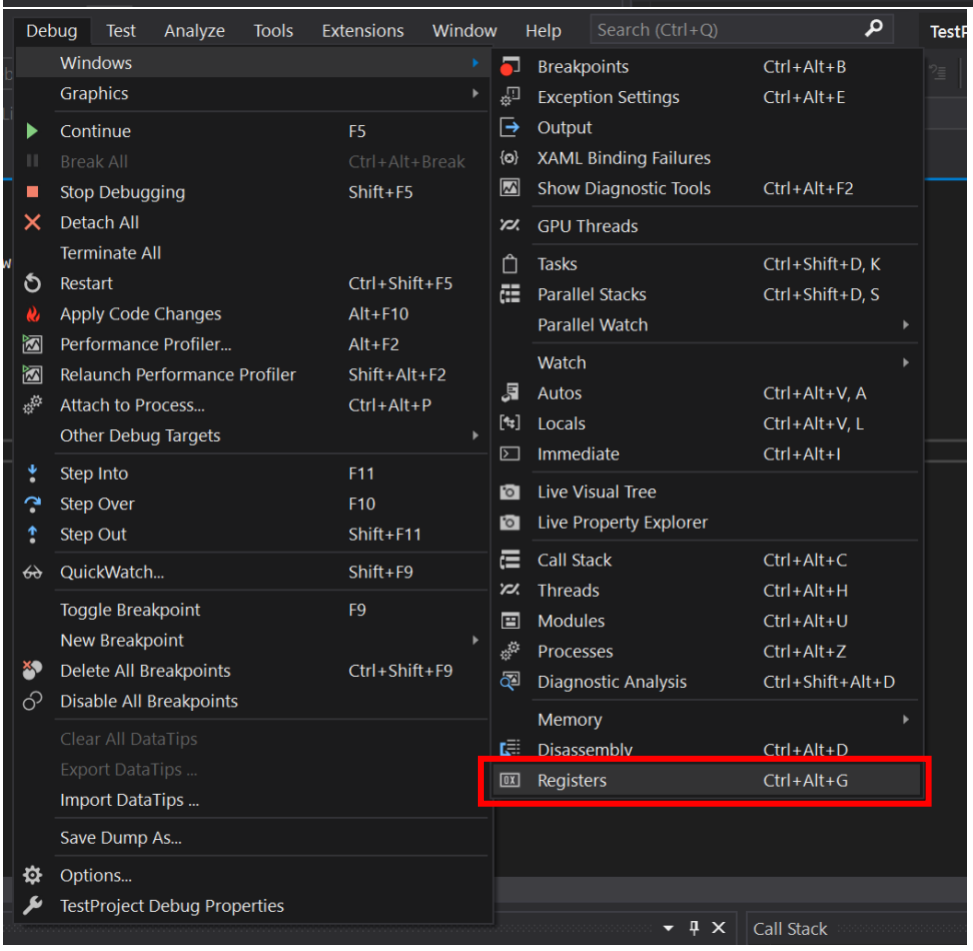
16. Add breakpoints wherever to want to halt and check the values of the registers/variables. Then start the debugging by clicking Debug -> Start Debugging



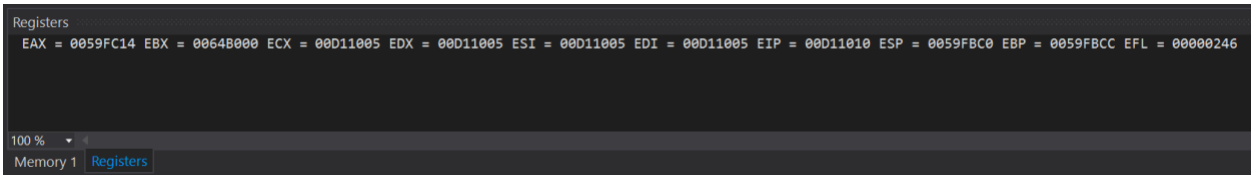
17. Once the debugging starts you see the diagnostic tools window and call stack windows. Now you need to enable the watch dog, memory and registers window as shown below:



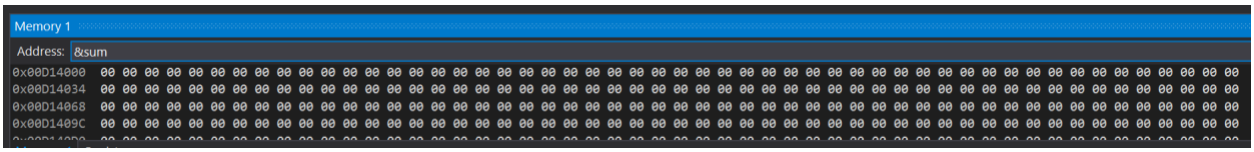

Debug
Test
Analyze
Tools
Extensions
Window
Help
Search (Ctrl+Q)
TestProject
Windows
Graphics
Continue F5
Break All Ctrl+Alt+Break
Stop Debugging Shift+F5
Detach All
Terminate All
Restart Ctrl+Shift+F5
Apply Code Changes Alt+F10
Performance Profiler... Alt+F2
Relaunch Performance Profiler Shift+Alt+F2
Attach to Process... Ctrl+Alt+P
Other Debug Targets
Step Into F11
Step Over F10
Step Out Shift+F11
QuickWatch... Shift+F9
Toggle Breakpoint F9
New Breakpoint
Delete All Breakpoints Ctrl+Shift+F9
Disable All Breakpoints
Clear All DataTips
Export DataTips ...
Import DataTips ...
Save Dump As...
Options...
TestProject Debug Properties
Breakpoints Ctrl+Alt+B
Exception Settings Ctrl+Alt+E
Output
XAML Binding Failures
Show Diagnostic Tools Ctrl+Alt+F2
GPU Threads
Tasks Ctrl+Shift+D, K
Parallel Stacks Ctrl+Shift+D, S
Parallel Watch
Watch
Autos Ctrl+Alt+V, A
Locals Ctrl+Alt+V, L
Immediate Ctrl+Alt+I
Live Visual Tree
Live Property Explorer
Call Stack Ctrl+Alt+C
Threads Ctrl+Alt+H
Modules Ctrl+Alt+U
Processes Ctrl+Alt+Z
Diagnostic Analysis Ctrl+Shift+Alt+D
Memory
Disassembly Ctrl+Alt+D
Registers Ctrl+Alt+G
Memory 1 Ctrl+Alt+M, 1
Memory 3 Ctrl+Alt+M, 3
Memory 4 Ctrl+Alt+M, 4
Call Stack


Debug
Test
Analyze
Tools
Extensions
Window
Help
Search (Ctrl+Q)
Windows
Graphics
Continue F5
Break All Ctrl+Alt+Break
Stop Debugging Shift+F5
Detach All
Terminate All
Restart Ctrl+Shift+F5
Apply Code Changes Alt+F10
Performance Profiler... Alt+F2
Relaunch Performance Profiler Shift+Alt+F2
Attach to Process... Ctrl+Alt+P
Other Debug Targets
Step Into F11
Step Over F10
Step Out Shift+F11
QuickWatch... Shift+F9
Toggle Breakpoint F9
New Breakpoint
Delete All Breakpoints Ctrl+Shift+F9
Disable All Breakpoints
Clear All DataTips
Export DataTips ...
Import DataTips ...
Save Dump As...
Options...
TestProject Debug Properties
Breakpoints Ctrl+Alt+B
Exception Settings Ctrl+Alt+E
Output
XAML Binding Failures
Show Diagnostic Tools Ctrl+Alt+F2
GPU Threads
Tasks Ctrl+Shift+D, K
Parallel Stacks Ctrl+Shift+D, S
Parallel Watch
Watch
Autos Ctrl+Alt+V, A
Locals Ctrl+Alt+V, L
Immediate Ctrl+Alt+I
Live Visual Tree
Live Property Explorer
Call Stack Ctrl+Alt+C
Threads Ctrl+Alt+H
Modules Ctrl+Alt+U
Processes Ctrl+Alt+Z
Diagnostic Analysis Ctrl+Shift+Alt+D
Memory
Disassembly Ctrl+Alt+D
Registers Ctrl+Alt+G
Call Stack

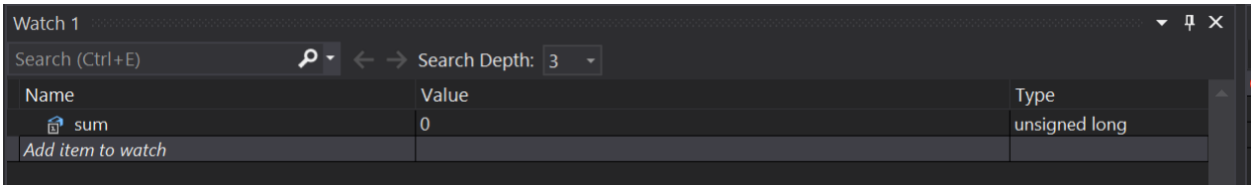
18. Now you see the corresponding tabs in your VS window:



You can check the values of different registers in the register window

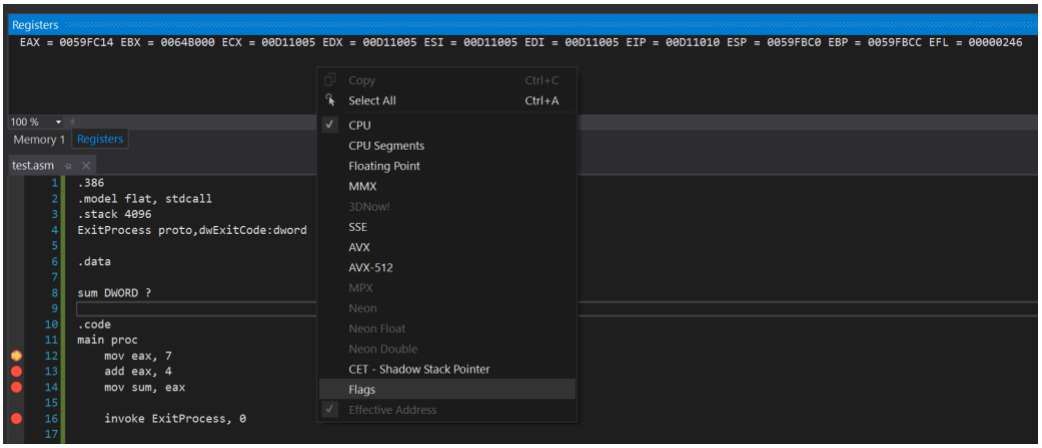


In the memory window, you can check the values of different variables by typing &variable-name in the address bar. Here we are looking for the sum variable so we give &sum.

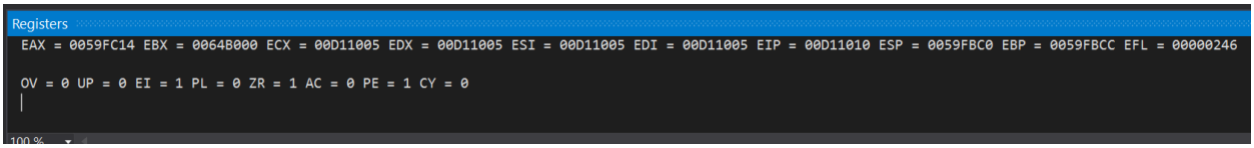


You can add variables to this watchdog to verify the values during your execution.

19. To enable the status flags in the registers window, right-click -> enable Flags



Now you can see the status flags as well:



20. Now click on continue to debug your code step by step according to your breakpoints. You can observe the values changing in the registers, memory and the watch windows.