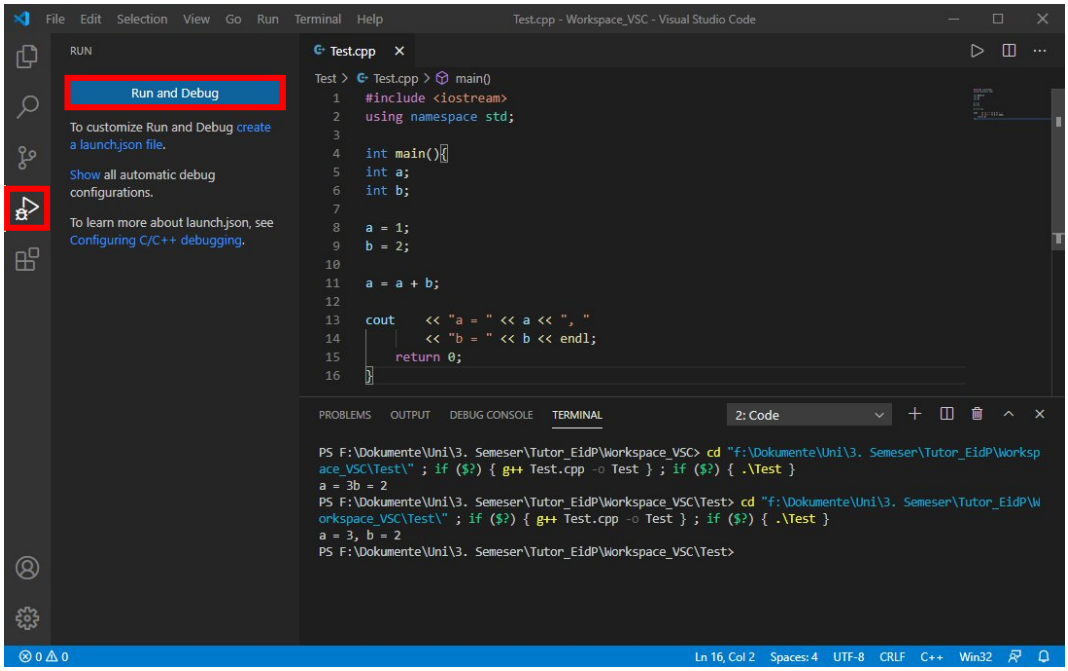
Visual Studio Code: Debugging

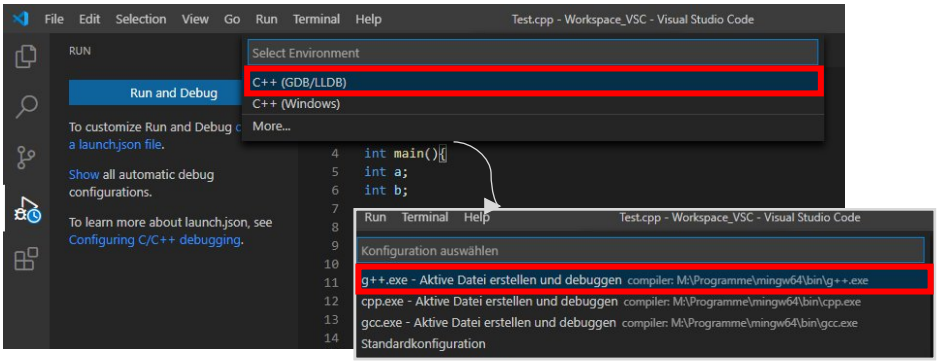
Ein Debugger ist ein Werkzeug zum Auffinden von *Bugs* (Fehlern) in Programmen. Dabei kann das Programm schrittweise ausgeführt und nach jedem Schritt der Zustand untersucht werden. Ein Schritt bezeichnet dabei immer eine Programmzeile und der Zustand eines laufenden Programms lässt sich (stark vereinfacht) durch die Variablenbelegung beschreiben. Der Debugger eignet sich **insbesondere** dazu, ein Programm zu untersuchen, das sich nicht wie vorgesehen verhält. Zögern Sie also bitte nicht, den Debugger einzusetzen.

Die folgende Anleitung beschreibt, wie der Debugger mit Visual Studio Code eingerichtet wird und wie der Prozess des „Debuggens“ funktioniert:

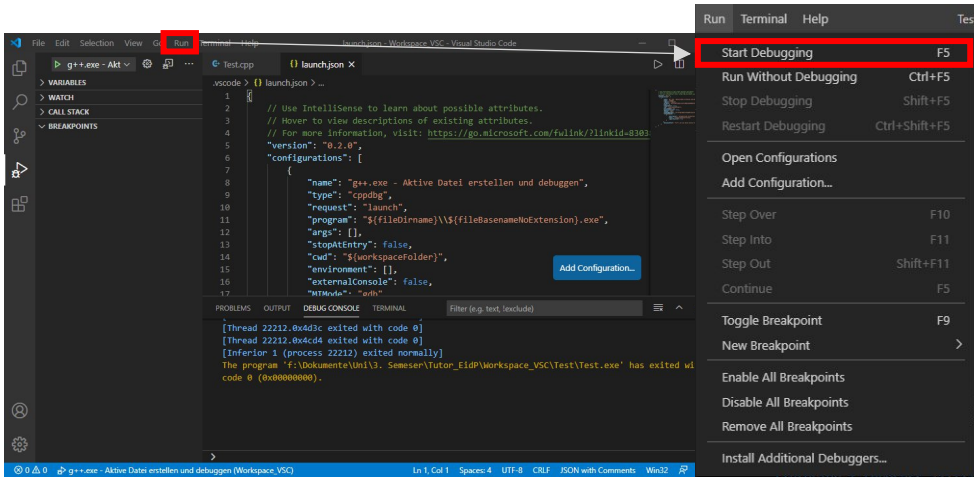
1. Auf der linken Seite auf das Debuggingsymbol und anschließend auf „Run and Debug“ klicken:



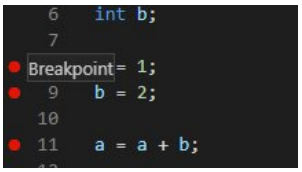
2. Nun erst „C++ (GDB/LLDB)“ und dann „g++.exe …“ auswählen

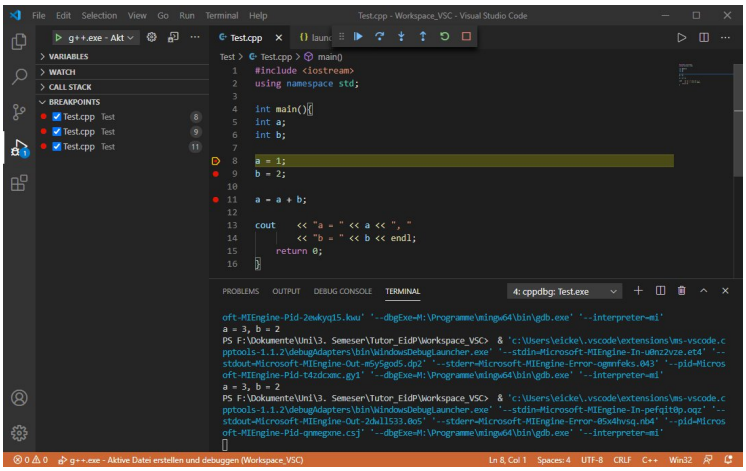


3. Wenn nun folgender Bildschirm angezeigt wird, kann über „Run“ und dann „Start Debugging“ der Debugger aufgerufen werden (alternativ direkt mit F5). Die Datei „launch.json“ kann im Anschluss wieder geschlossen werden.



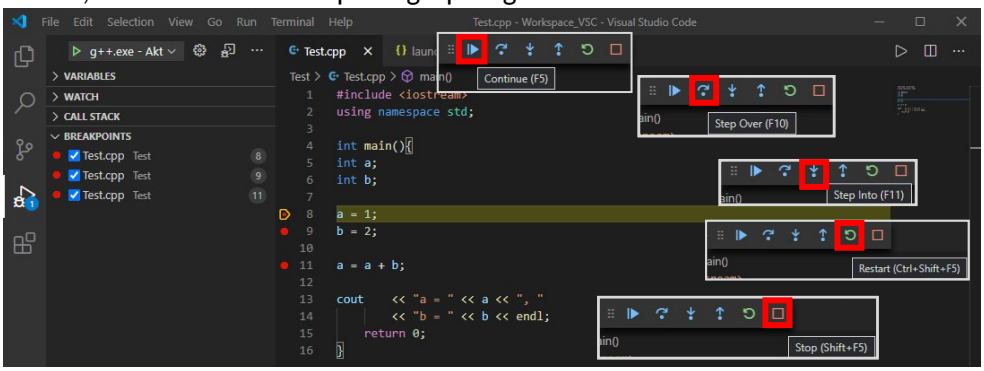
4. Mittels Linksklick neben eine Codezeile kann in dieser ein Breakpoint gesetzt werden





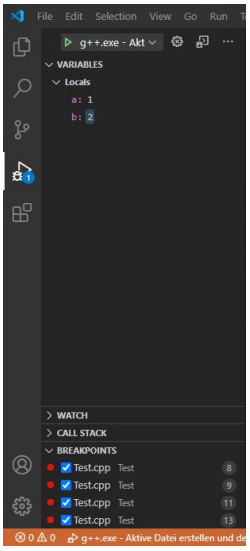
5. Wenn der Debugger nach dem Setzen der Breakpoints gestartet wird, hält dieser an dem ersten Breakpoint an.

6. Über das oben in der Mitte liegende Kontrollzentrum kann nun mit dem linken Button, oder der Taste F5, zum nächsten Breakpoint gesprungen werden.

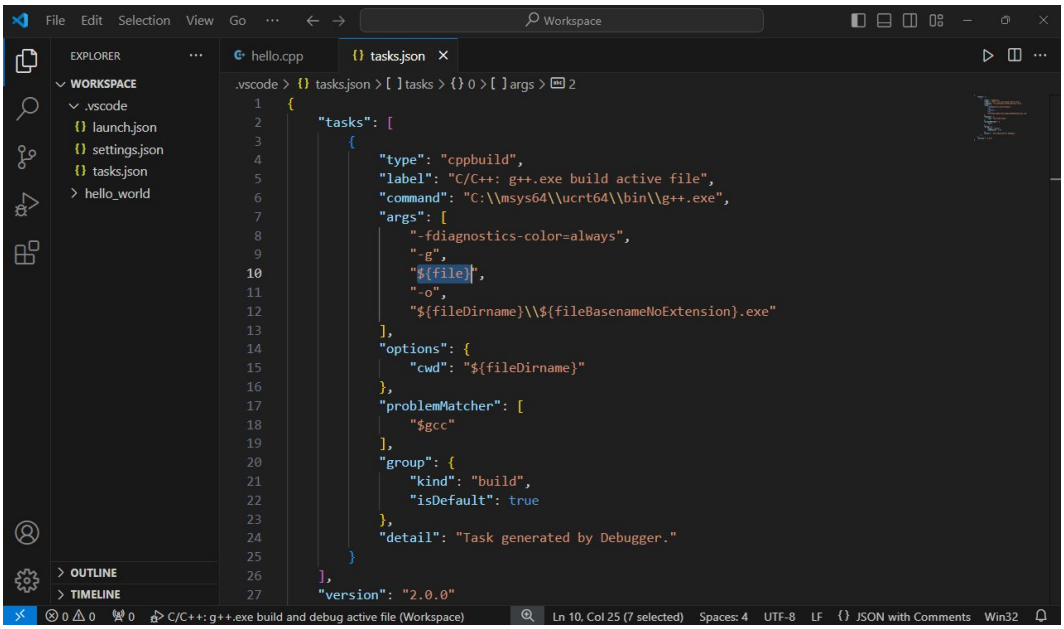


Mit den Knöpfen „Step Over“ (F10) bzw. „Step Into“ (F11) kann der Code zeilenweise abgelaufen werden. Wobei mit dem Knopf „Step Into“ in eine auszuführende Funktion gesprungen werden kann. „Stop“ (Shift+F5) stoppt den Code. „Restart“ startet das Programm erneut.

7. Auf der linken Seite kann im Debugging-Menü nun unter dem Reiter „VARIABLES“ der Wert jeder Variable eingesehen werden. Der dort angegebene Wert entspricht dem Wert **vor** der aktuellen Codezeile.



8. Sollte ihr Programm aus mehreren Dateien bestehen müssen sie die Kompilation analog zu Schritt E7 aus der initialen Anleitung anpassen. Öffnen sie dazu die Datei „.vscode/tasks.json“ und suchen sie den Eintrag, welcher für den Aufruf von g++ zuständig ist.



Ändern sie in der Liste „args“ den Einrag **`${file}`** zu **`${fileDirname}/*.cpp`**
Dies führt dazu, das alle Dateien im selben Verzeichnis wie die ausgewählte Datei gemeinsam übersetzt werden.

Abschließend noch ein paar allgemeine Hinweise:

- Achten Sie immer darauf, dass Sie Ihr Programm gespeichert haben, bevor Sie den Debugger starten.
- Setzen Sie bitte keine Breakpoints in leeren Zeilen.
- Wenn Sie keine Breakpoints gesetzt haben, läuft das Programme ohne Anzuhalten durch. Daher sollten Sie, wenn Sie das Programm untersuchen möchten, auch einen Breakpoint setzen.