

Erstellung der  
Anwendung zum Einlesen der  
Lotsenkommandos, die aus mehreren Dateien  
besteht, für verschiedene Plattformen

## Integration mit cmake



CMake ist ein plattformunabhängiges Programmierwerkzeug für die Entwicklung und Erstellung von Software. CMake wurde als Teil des Visible Human Project für das „Insight Segmentation and Registration Toolkit“ erstellt

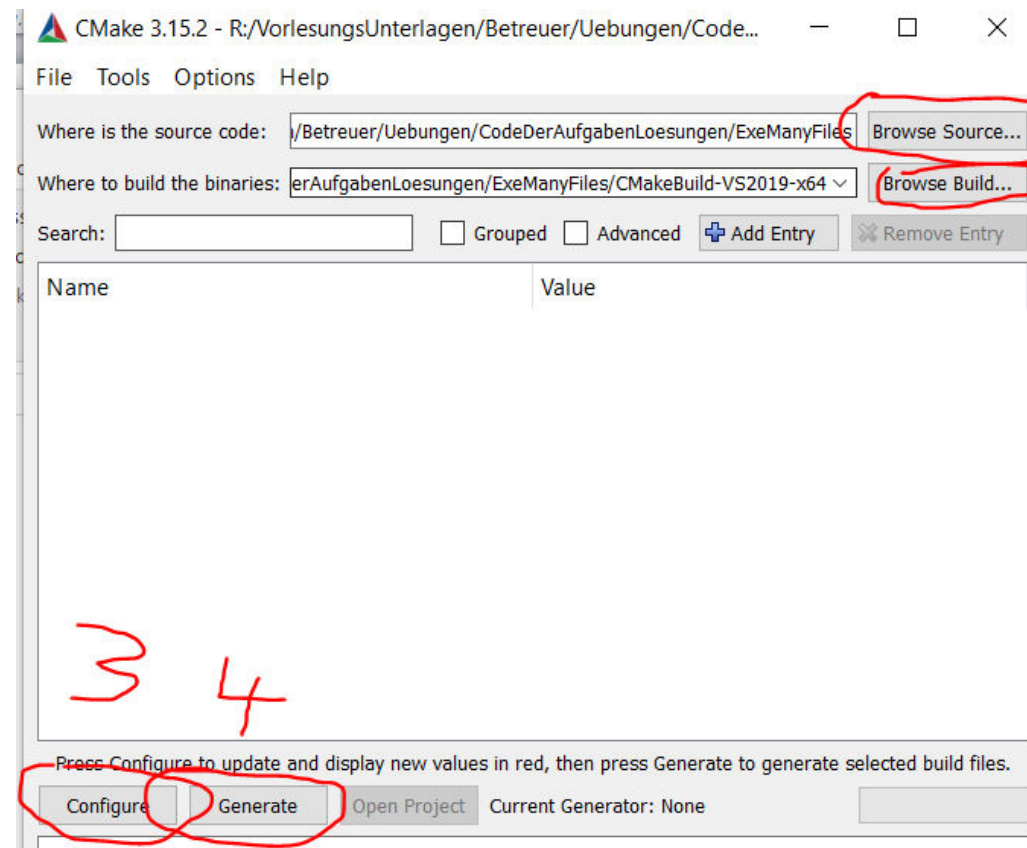
Download: <https://cmake.org/download/>  
und dann installieren

# Erzeugung des Projektes mit CMake



Die folgenden Schritte funktionieren auf allen Systemen, auf denen Sie CMake installieren konnten (Windows, Linux, Mac ...)

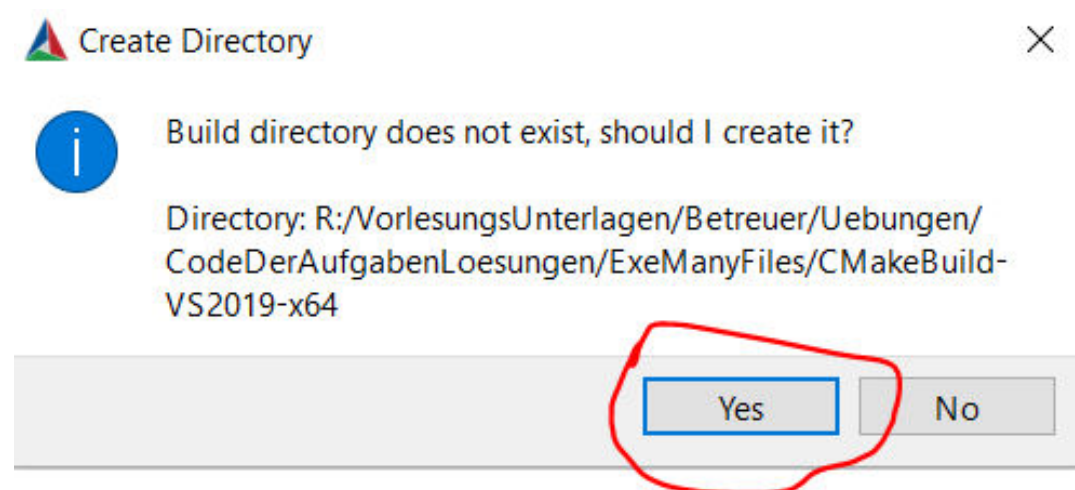
1. Rufen Sie CMake über die GUI (unter Windows) auf.



## Erzeugung des Projektes mit Cmake (



1. Browsen Sie zum Ordner, in dem sich der Code mit der von mir bereitgestellten Datei CMakeList.txt befindet.
2. Browsen Sie in den Ordner, in dem Sie die Projektdateien installieren wollen (der gleiche Ordner wie in 1, aber ein (noch nicht vorhandener) Unterordner davon wird empfohlen, z.B. CMakeBuild-VS2019-x64 (für Visual Studio 2019 als 64 Bit-Anwendung)
3. Klicken Sie auf Configure und bestätigen Sie, dass ein neuer Ordner erstellt werden soll.



## Erzeugung des Projektes mit Cmake (3)



und wählen Sie hier die Entwicklungsumgebung/Compiler aus, für den Sie Code erzeugen wollen (üblicherweise Visual Studio 18 2022). Sie können auch noch auswählen, ob Sie eine x64 oder win32 Anwendung erstellen wollen

Klicken Sie dann auf fertig stellen (Finish). Jetzt kann es schon mal 60 Sekunden dauern, bis alle Einstellungen gefunden sind.

Specify the generator for this project  
Visual Studio 17 2022

Optional platform for generator (if empty, generator uses: x64)  
x64

Optional toolset to use (argument to -T)  
[Empty field]

Use default native compilers  
 Specify native compilers  
 Specify toolchain file for cross-compiling  
 Specify options for cross-compiling

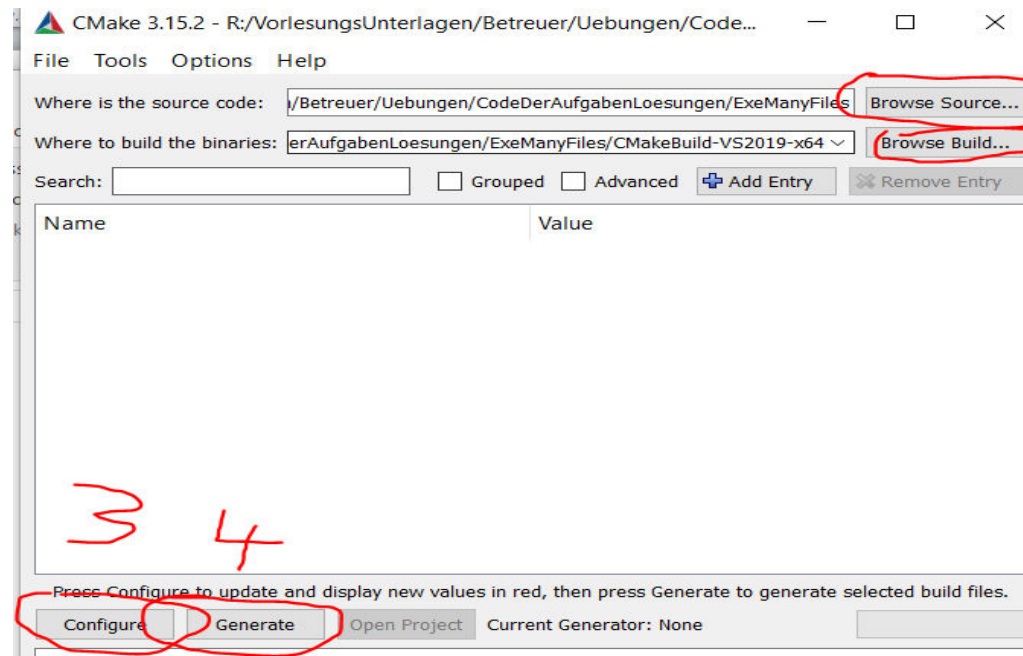
„Configuration Done“ sollte im unteren Bereich stehen und mit mittleren gibt es zwei neue Variablenwerte in Rot

NAME	VALUE
CMAKE_CONFIGURATION_TYPES	Debug;Release;MinSizeRel;RelWithDebInfo
CMAKE_INSTALL_PREFIX	C:/Program Files (x86)/ExeManyFiles

## Erzeugung des Projektes mit Cmake (4)

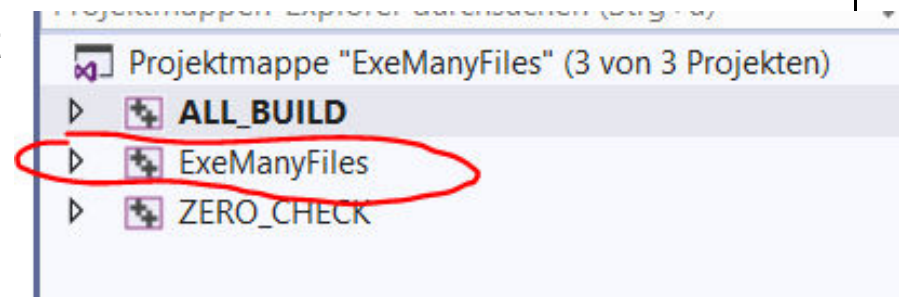
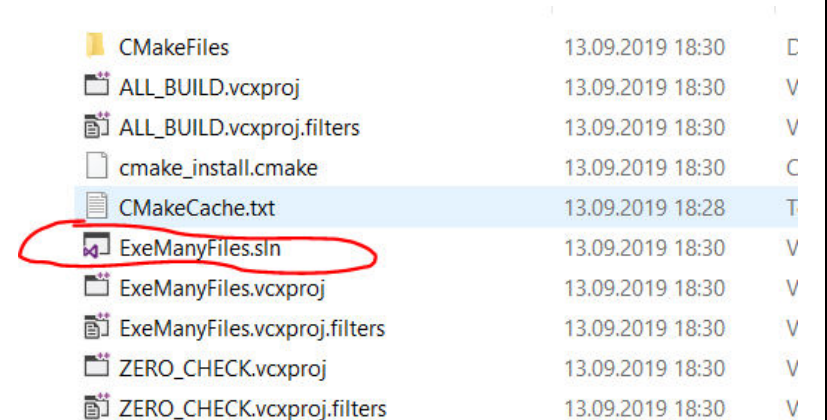


4. Zum Schluss klicken Sie noch auf „Generate“. Das geht im Allgemeinen sehr schnell.



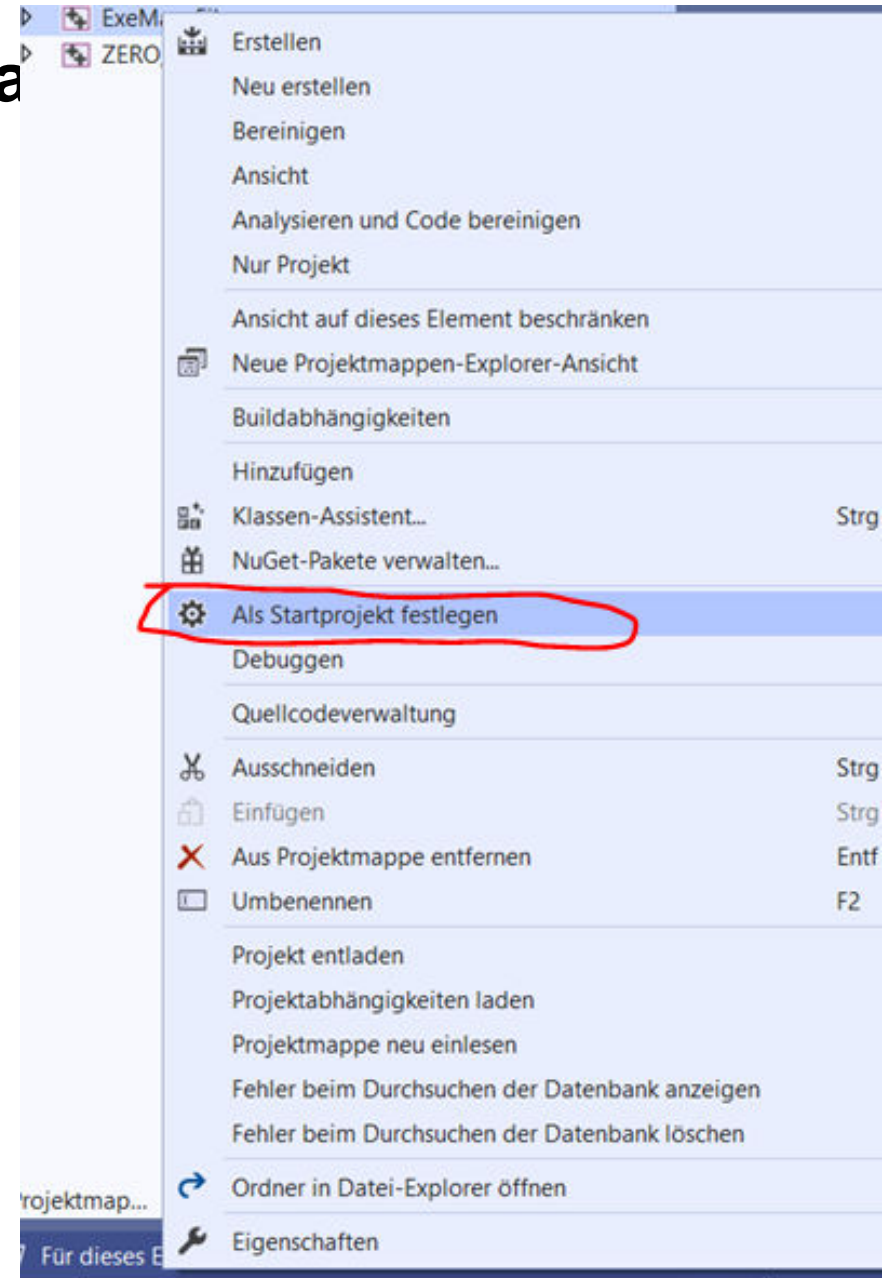
## Erstellen des Projektes mit Visual Studio

1. Gehen Sie in den Ordner, in dem sich die von cmake erstellten Dateien befinden (den Sie unter 2 ausgewählt haben)
2. und führen einen Doppelklick auf die Projektdatei mit der Endung sln aus. (Gilt nur für Visual Studio, bei anderen Umgebungen werden andere Konfigurationsdateien erstellt)
3. Visual Studio sollte sich (nach einiger Zeit öffnen) öffnen  
Legen Sie ExeManyFiles (oder anderer Name) durch Rechtsklick als Startprojekt fest.



## Erstellen des Projektes mit Visual

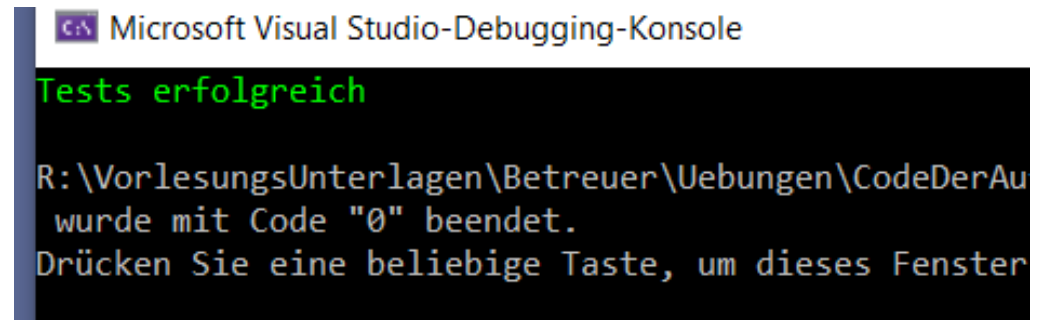
3. Legen Sie ExeManyFiles durch Rechtsklick als Startprojekt fest





## Kompilierung des Projektes mit Visual Studio (2)

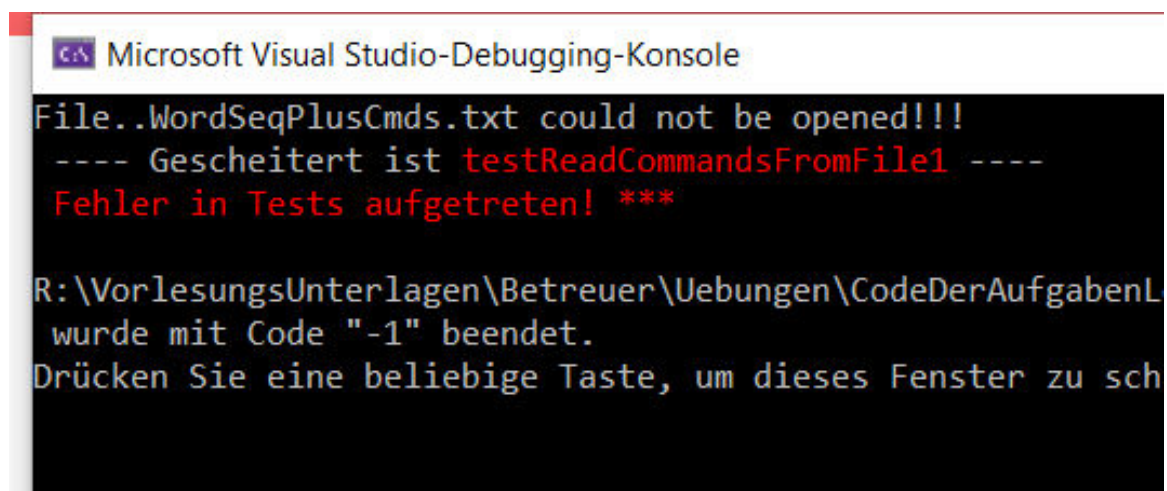
1. Erstellen Sie das Projekt  
(entweder im Menü  
auswählen oder F7  
drücken)
2. „Erneut laden“ drücken
3. Wenn Sie eine Fehlermeldung erhalten,  
ging etwas schief, was eigentlich nicht sein  
sollte
4. Führen Sie das Programm  
aus (Strg-F5)



```
Microsoft Visual Studio-Debugging-Konsole
Tests erfolgreich
R:\Vorlesungsunterlagen\Betreuer\Uebungen\CodeDerAu
wurde mit Code "0" beendet.
Drücken Sie eine beliebige Taste, um dieses Fenster
```

## Probleme

Falls Ihr Programm mit der folgenden Fehlermeldung scheitert, wird es unter Windows daran liegen, dass Sie die Build-Datei an einer anderen als der vorgeschlagenen Stelle (direkt in einem Unterordner des Codeordners abgelegt haben).

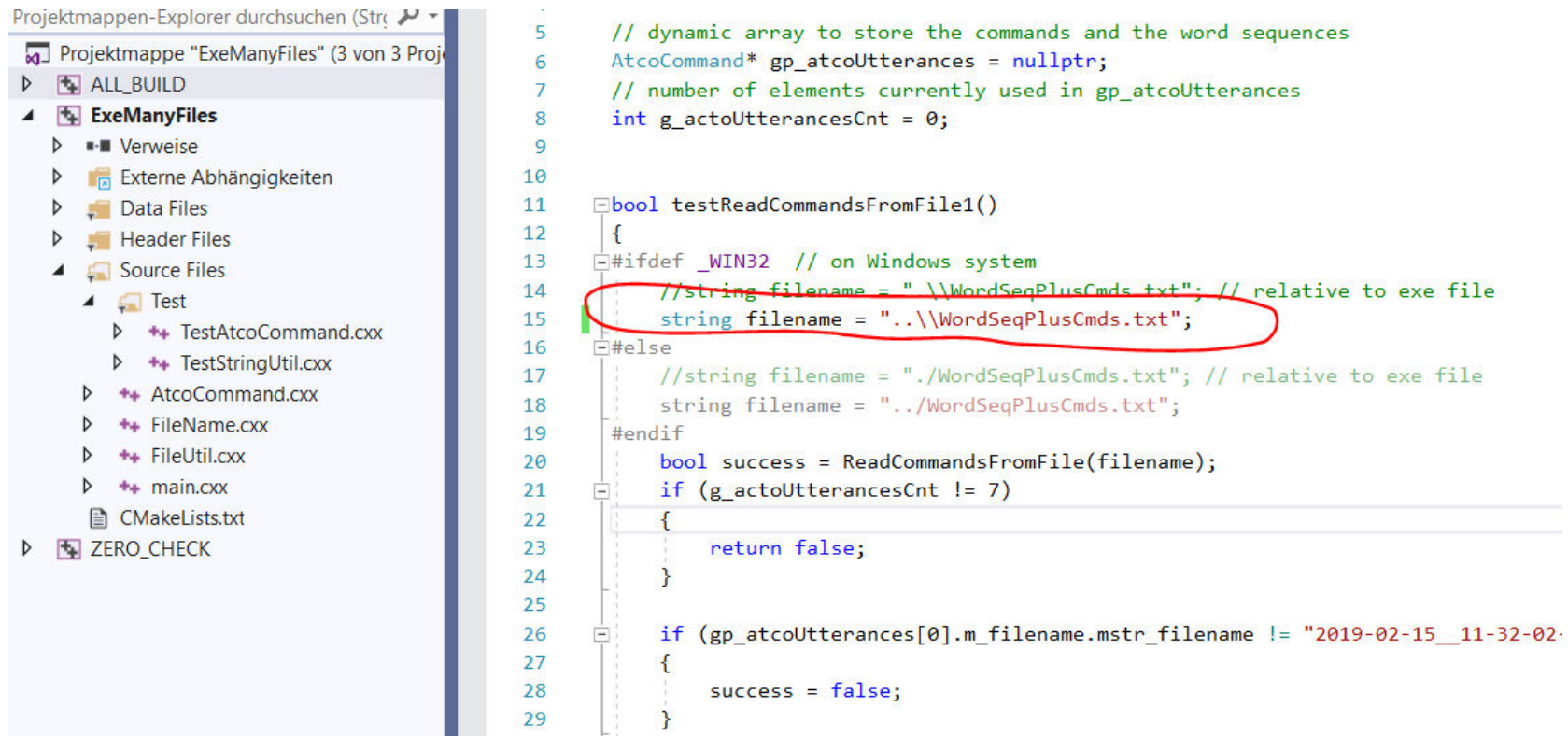


```
Microsoft Visual Studio-Debugging-Konsole
File..WordSeqPlusCmds.txt could not be opened!!!
---- Gescheitert ist testReadCommandsFromFile1 ----
Fehler in Tests aufgetreten! ***

R:\Vorlesungsunterlagen\Betreuer\Uebungen\CodeDerAufgabenL
wurde mit Code "-1" beendet.
Drücken Sie eine beliebige Taste, um dieses Fenster zu sch
```

## Mögliche Problemlösung

Versuchen Sie dann den Pfad der Textdatei anzupassen (z.B. „..\“ statt „..\“ oder „..\..\“). Suchen Sie dazu die erstellte Anwendung (exe-Datei unter Windows) und finden Sie die relative Lage der Anwendung zur Textdatei WordSeqPlusCmds.txt heraus.



```
1 // dynamic array to store the commands and the word sequences
2 AtcoCommand* gp_atcoUtterances = nullptr;
3 // number of elements currently used in gp_atcoUtterances
4 int g_actoUtterancesCnt = 0;
5
6
7
8
9
10
11 bool testReadCommandsFromFile1()
12 {
13 #ifdef _WIN32 // on Windows system
14 //string filename = "\\WordSeqPlusCmds.txt"; // relative to exe file
15 string filename = "../WordSeqPlusCmds.txt";
16 #else
17 //string filename = "./WordSeqPlusCmds.txt"; // relative to exe file
18 string filename = "../WordSeqPlusCmds.txt";
19 #endif
20 bool success = ReadCommandsFromFile(filename);
21 if (g_actoUtterancesCnt != 7)
22 {
23 return false;
24 }
25
26 if (gp_atcoUtterances[0].m_filename.mstr_filename != "2019-02-15_11-32-02-
27 {
28 success = false;
29 }
```