

# Компьютерная графика

## Лабораторная работа №0.

### Настройка ИСР на примере VSCodium для Windows

Быковских Д.А.

сентябрь 2024 г.

## 1 Введение

Данная инструкция содержит подробное описание установки и настройки интегрированной среды разработки (ИСР или IDE) на примере ИСР VSCodium (далее VSCodium), а также других вспомогательных инструментов и приложений, необходимых для успешной разработки ваших приложений по дисциплине "Компьютерная графика", включая компиляцию, запуск и отладку. Эту инструкцию можно также использовать для установки и настройки Visual Studio Code (VS Code) или Code – OSS.

Следует подчеркнуть, что на сегодняшний день мне **неизвестен** способ корректной работы приложений, использующих графические библиотеки, **в виртуальной машине**. (Если у кого-то вдруг получится, я готов перенять опыт.)

## 2 Настройка ИСР

Для того чтобы разрабатывать программы под ОС семейства MS Windows (далее Windows) необходимо выполнить ниже описанные шаги.

### 2.1 Установка среды MSYS2

MSYS2 — среда разработки и исполнения программ для Windows. Она включает в себя командную оболочку и утилиты, которые позволяют разработчикам работать с программами и библиотеками, предназначенными для Unix-подобных операционных систем, под управлением Windows.

Перейдите по [ссылке](#) и скачайте установочный файл, установите MSYS2 (там даже показано, как устанавливать пакеты).

## 2.2 Установка с помощью командной оболочки MSYS2 необходимых библиотек, компиляторов и др.

После установки запустите командную оболочку (terminal) MSYS2, а затем выполните следующие разбитые на группы команды.

### 1. Обновите базовые пакеты

```
расман -Syu --noconfirm
```

### 2. Установите пакеты, связанные со сборкой и компиляцией приложения

```
расман -S --noconfirm mingw-w64-x86_64-clang  
расман -S --noconfirm mingw-w64-x86_64-gcc  
расман -S --noconfirm mingw-w64-x86_64-gdb  
расман -S --noconfirm mingw-w64-x86_64-cmake
```

### 3. Установите пакеты, связанные с графикой

```
расман -S --noconfirm mingw-w64-x86_64-glfw  
расман -S --noconfirm mingw-w64-x86_64-glew  
расман -S --noconfirm mingw-w64-x86_64-assimp  
расман -S --noconfirm mingw-w64-x86_64-glm  
расман -S --noconfirm mingw-w64-x86_64-soil  
расман -S --noconfirm mingw-w64-x86_64-stb  
расман -S --noconfirm mingw-w64-x86_64-libpng
```

### 4. Установите пакеты, связанные с Vulkan API

```
расман -S --noconfirm mingw-w64-x86_64-vulkan-devel  
расман -S --noconfirm mingw-w64-x86_64-glslang
```

Обратите внимание, что первая среди этих групп команда связана с обновлением. Возможно, потребуется ее выполнить еще раз, поскольку в процессе обновления может быть необходим перезапуск командной оболочки, прежде чем действительно установятся все обновления.

Примечание. По умолчанию исполняемые бинарные файлы, имеющие расширение exe, находятся в директории "C:\msys64\mingw64\bin". Говоря в целом, библиотеки, заголовочные файлы и пр. расположены в поддиректориях директории "C:\msys64\mingw64".

## 2.3 Установка VSCodium

Скачайте с [официального сайта](#) установочный файл, имеющий расширение msі или exe, и установите VSCodium.

Примечание. Эта ИСП — аналог VS Code, является свободной и не передает стороннюю информацию (например, телеметрию) компании Microsoft.

## 2.4 Установка расширений для VSCodium

Запустите VSCodium и перейдите на вкладку Extensions (расширения) с помощью комбинации клавиш Ctrl+Shift+X, чтобы установить расширения.

Теперь установите следующие расширения (см. рис. 1):

- MSYS2/Cygwin/MinGW/Clang support;
- CMake Tools;
- CMake;
- C/C++ ([ССЫЛКА](#));
- glsl-canvas ([ССЫЛКА](#));
- Shader languages support for VS Code;
- vscode-pdf.

Недостающие расширения, т.е. VSIX файлы, можно найти на [официальном сайте Marketplace](#), где размещаются расширения.

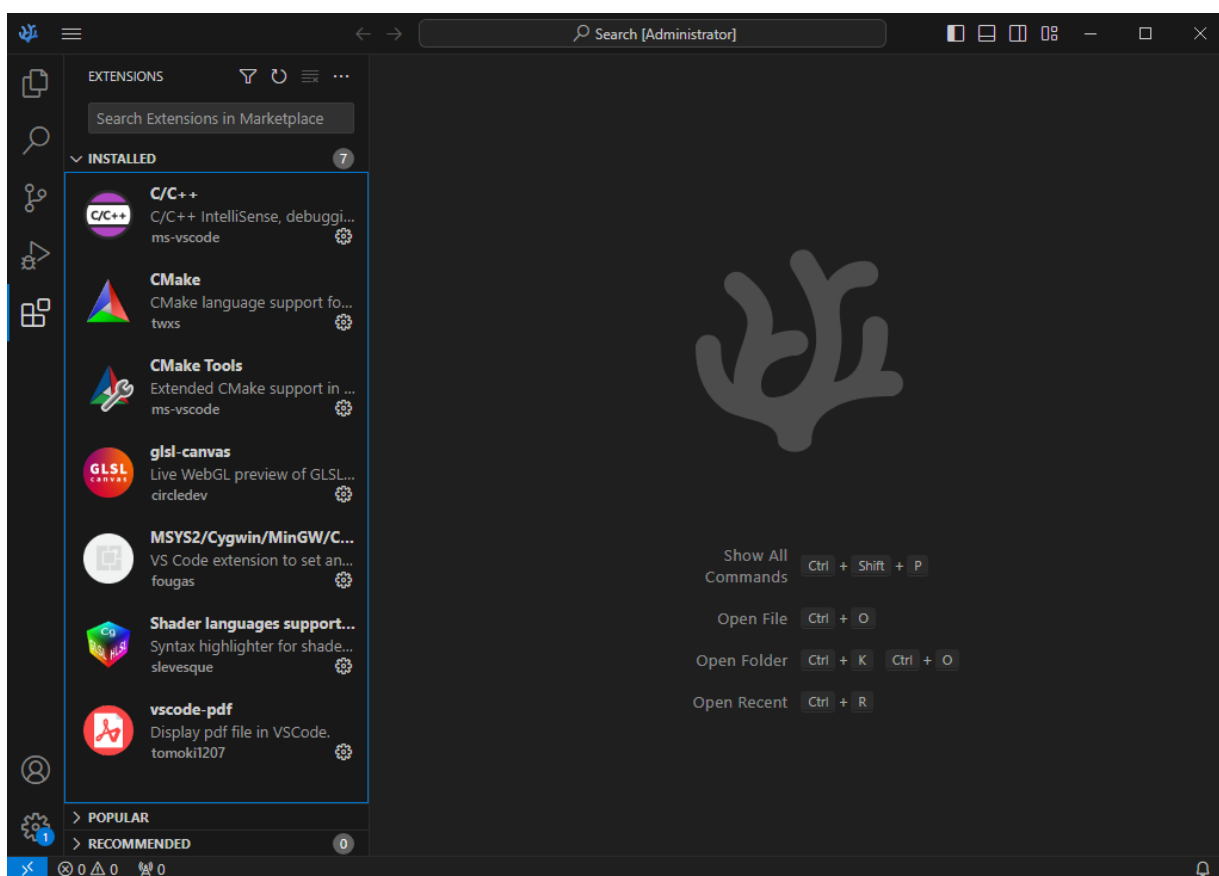


Рис. 1: Отображение установленных расширений

Например, чтобы установить glsl-canvas расширение, необходимо скачать VSIX файл, а для этого необходимо кликнуть на ссылку "Download Extension" на [странице сайта](#) (см. правый нижний угол рис. 2).

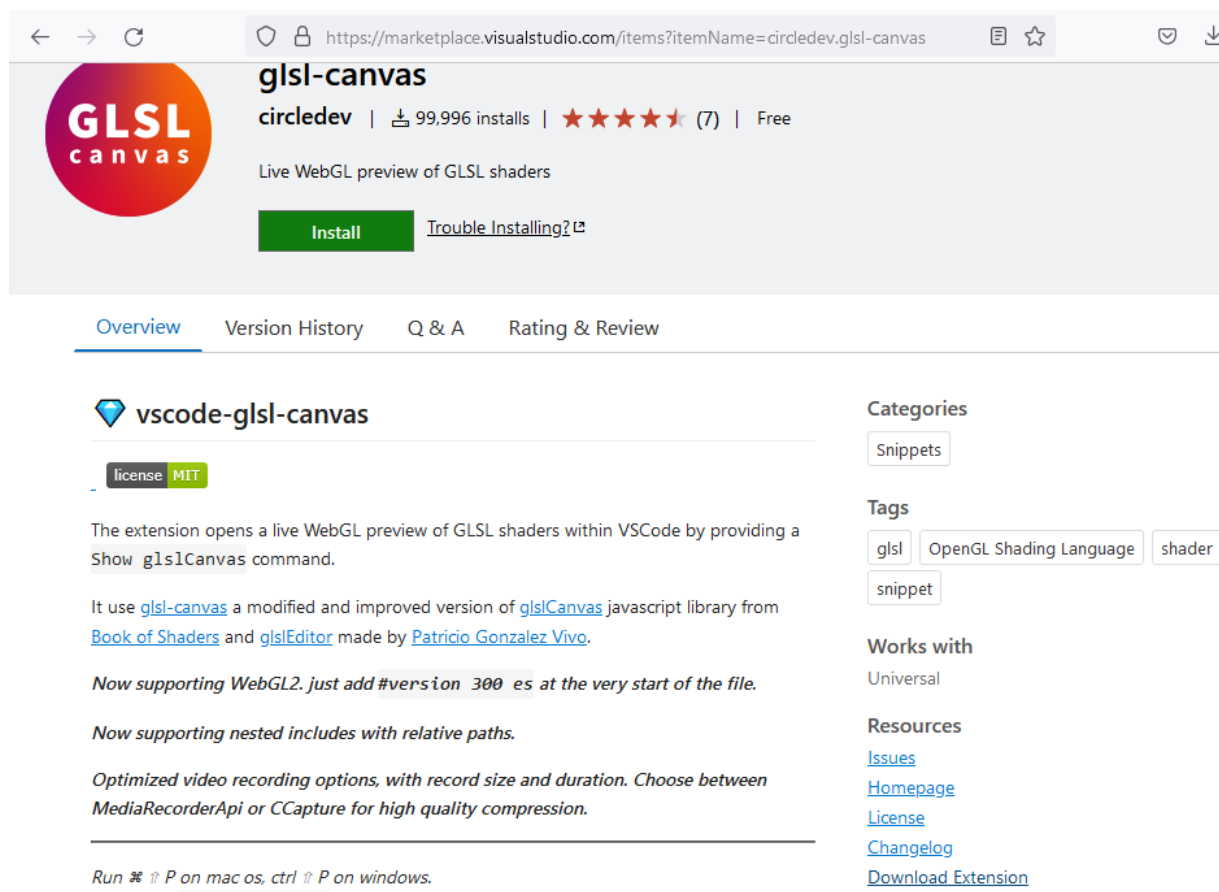


Рис. 2: Скачивание glsl-canvas расширения

Некоторые расширения, после их скачивания, можно установить следующим образом. На панели в правом верхнем углу необходимо навести курсор на "3 точки" и нажать клавишу мыши, после выбрать пункт "Install from VSIX" (см. рис. 3).

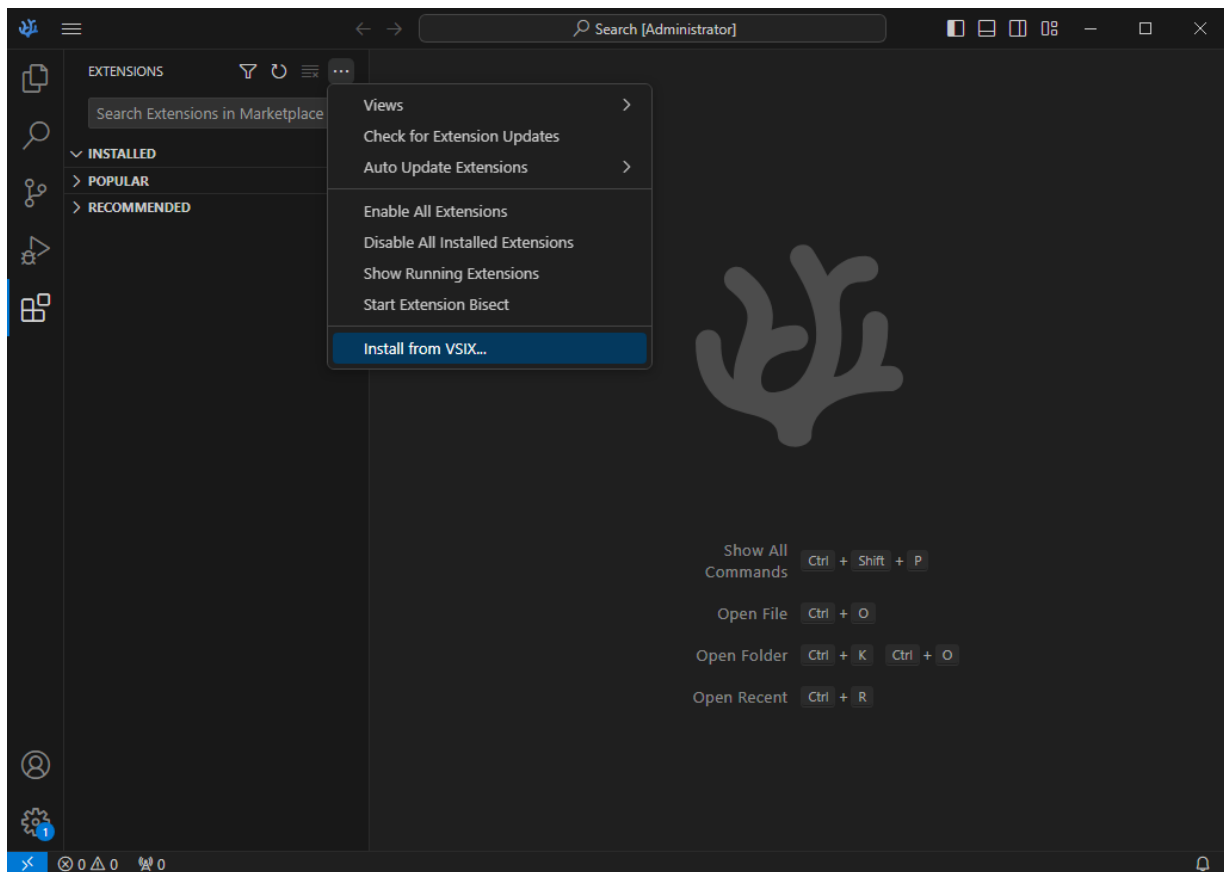


Рис. 3: Установка скачанных расширений

## 2.5 Настройка параметров VSCodium

Зайдите в настройки, например, с помощью комбинации клавиш `Ctrl+,` (эту запятую не так просто увидеть) и измените параметр `"cmake.cmakePath"` (этот параметр можно написать в строке поиска для быстрого поиска) на следующий `"C:\msys64\mingw64\bin\cmake.exe"` (см. рис. 4) **или** добавьте в файл `"settings.json"` следующую строку (см. рис. 5):

```
"cmake.cmakePath": "C:\\msys64\\mingw64\\bin\\cmake.exe",
```

**Обратите внимание,** что если выбрана вкладка `User`, то изменяются настройки для пользователя, т.е. для всех проектов, а если выбрана вкладка `Workspace`, то только для текущего проекта и изменения вносятся в файл `".vscode/settings.json"`.

Примечание. В интерфейсе VSCodium и в файле `"settings.json"` с настройками `\` (backslash) отображается по-разному.

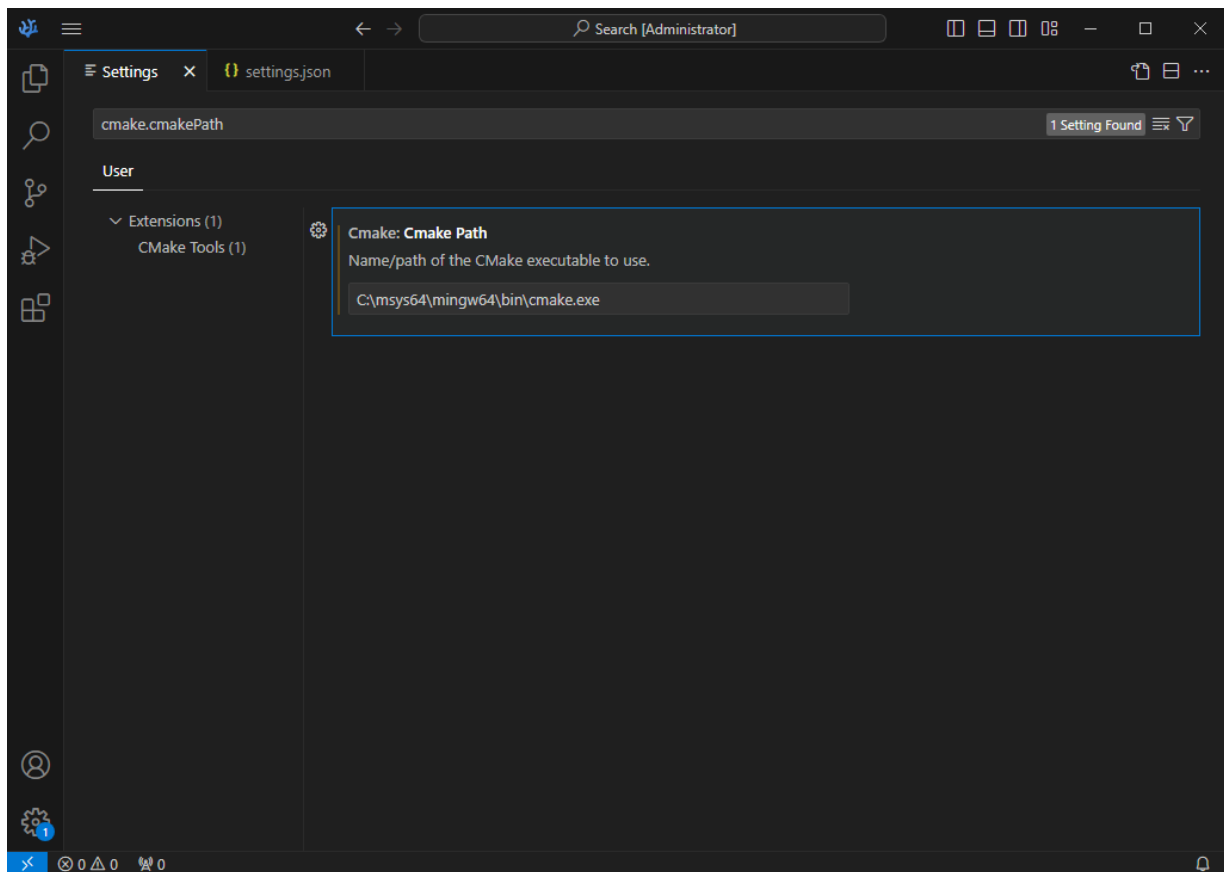


Рис. 4: Изменение настроек с помощью интерфейса

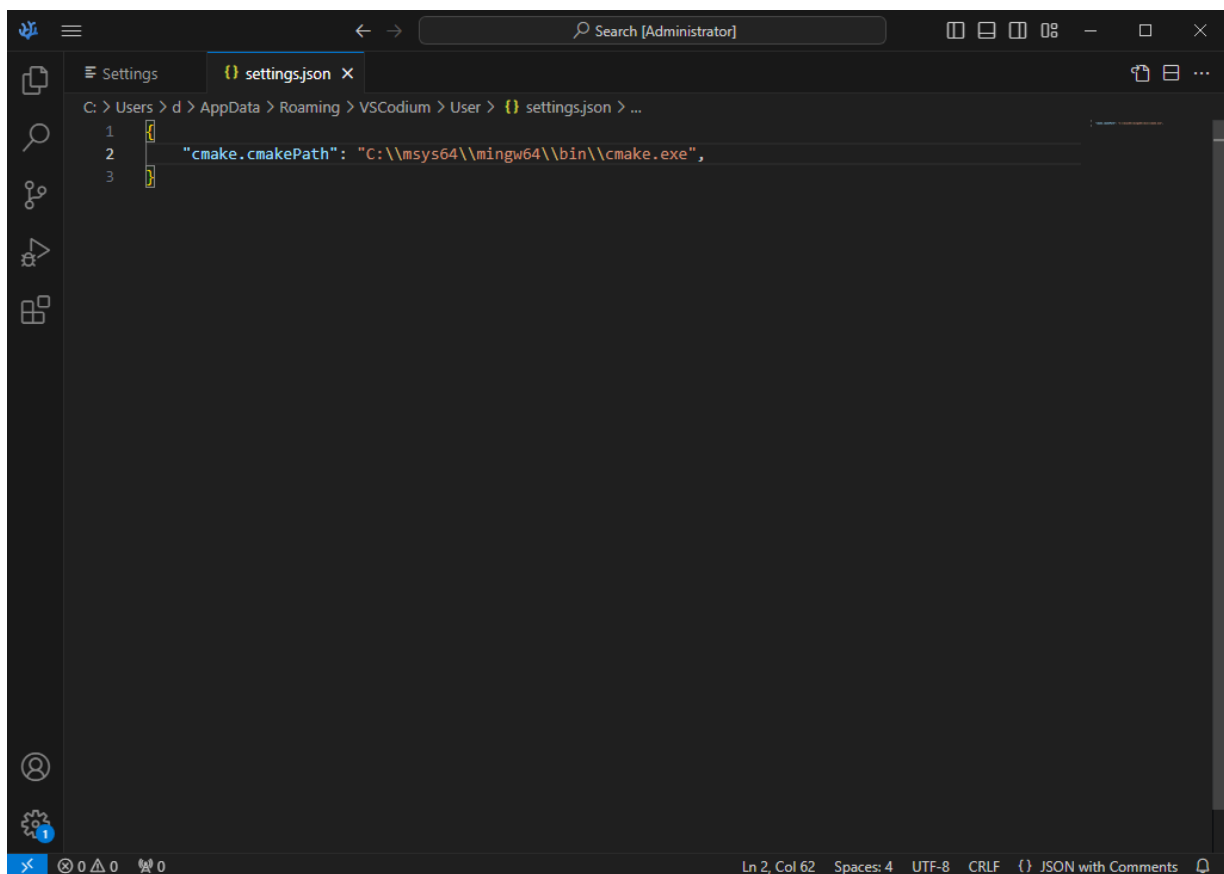


Рис. 5: Изменение настроек в файле "settings.json"

### 3 Запуск проекта "P2-Geometric\_primitives"

Чтобы убедиться, что все работает корректно, осталось лишь скомпилировать и запустить проект.

Но перед этим откройте в новом окне VSCode, нажав комбинацию клавиш Ctrl+Shift+N. После нажмите Ctrl+O, и выберите в директорию Practicum/P2-Geometric\_primitives.

Теперь, чтобы использовать файл (.vscode/cmake-kits.json), расположенный в скрытой директории, с прописанными путями к необходимым установленным вами компиляторам, выполните комбинацию клавиш Ctrl+Shift+P, выберите ">CMake: Select a Kit"(см. рис. 6) и сразу из выпадающего списка выберите "gcc and g++"(см. рис. 7).

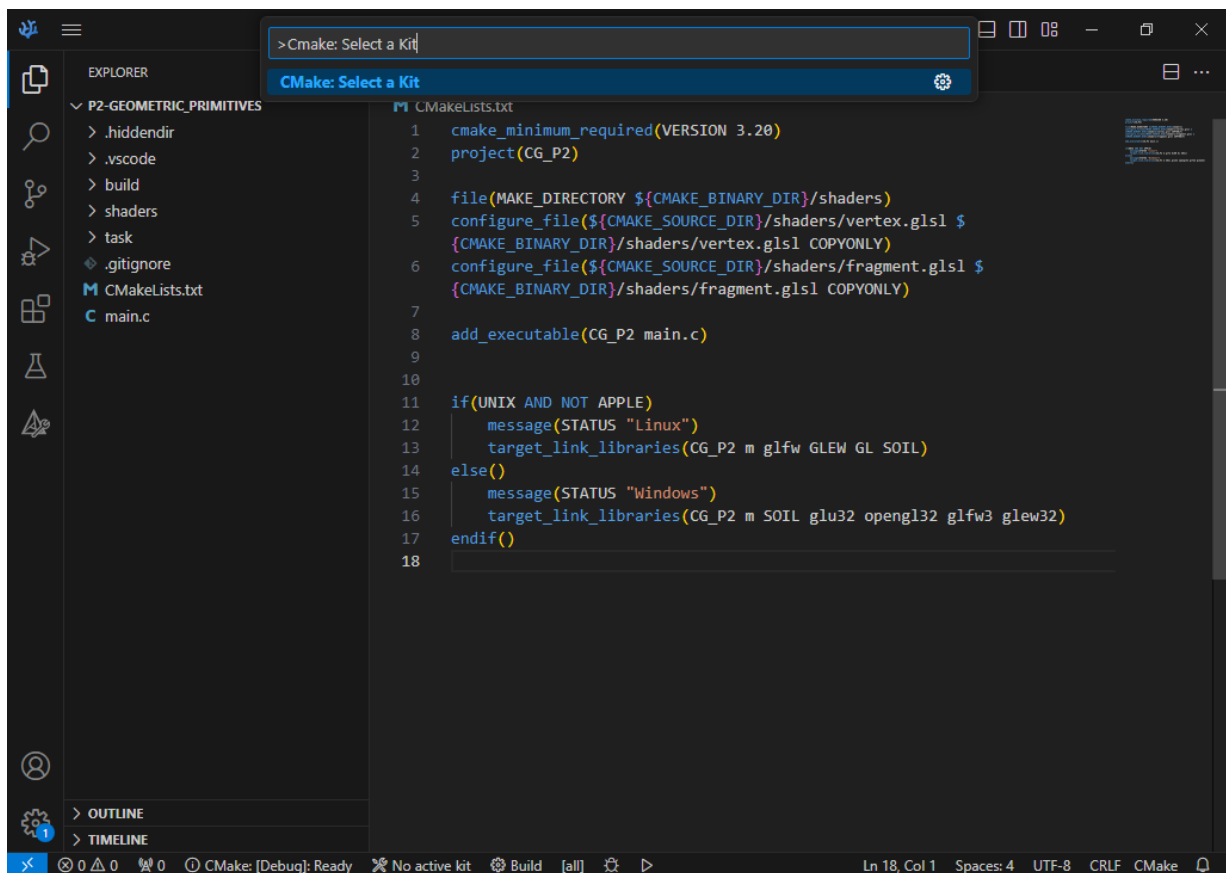


Рис. 6: Настройка CMake

Примечание. Возможно, могут возникнуть проблемы, если в пути, т.е. адресе директории, куда скопирован репозиторий <https://github.com/sci-d3v/CG.git>, содержатся русские символы.

Откройте файл CMakeLists.txt и сохраните его, нажав Ctrl+S, при этом произойдет автоматическая сбока или пересборка проекта.

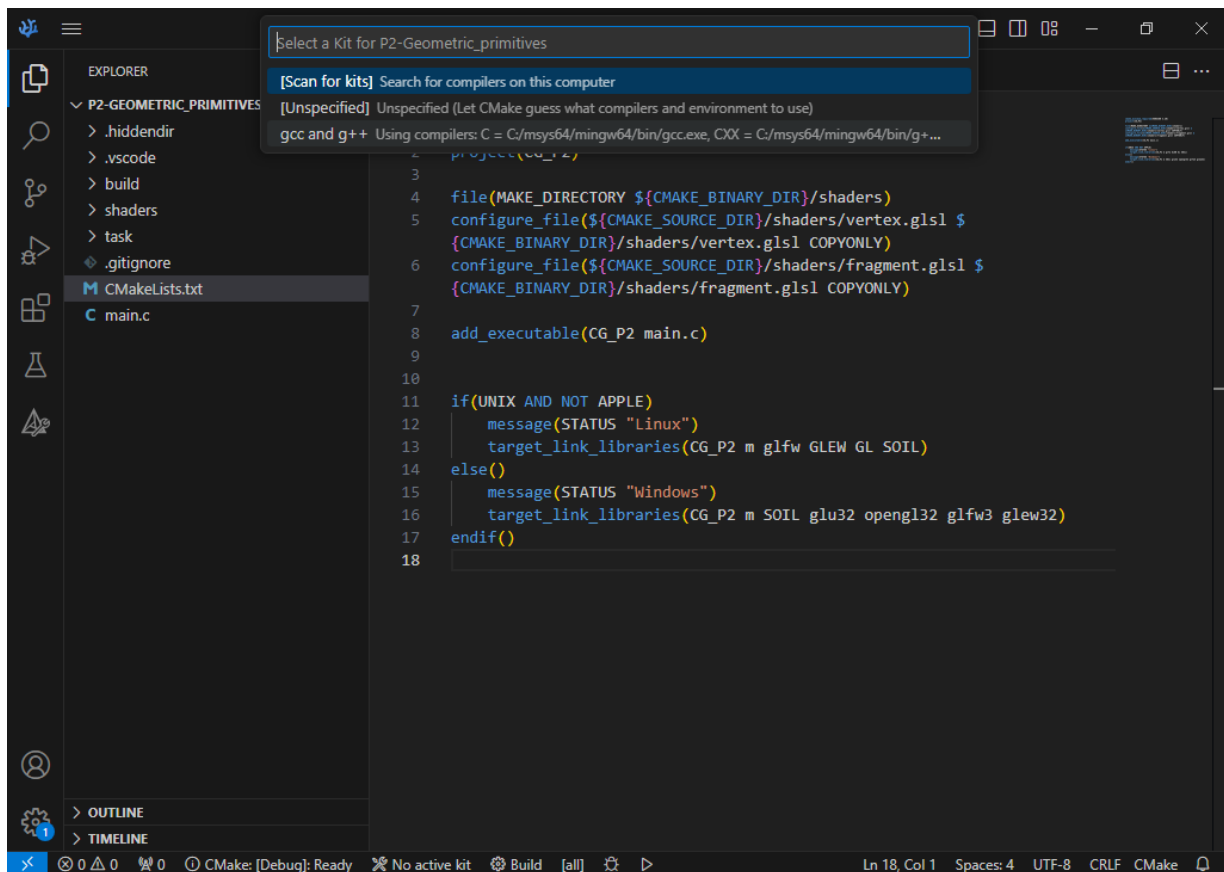


Рис. 7: Возможные варианты путей к компиляторам для CMake

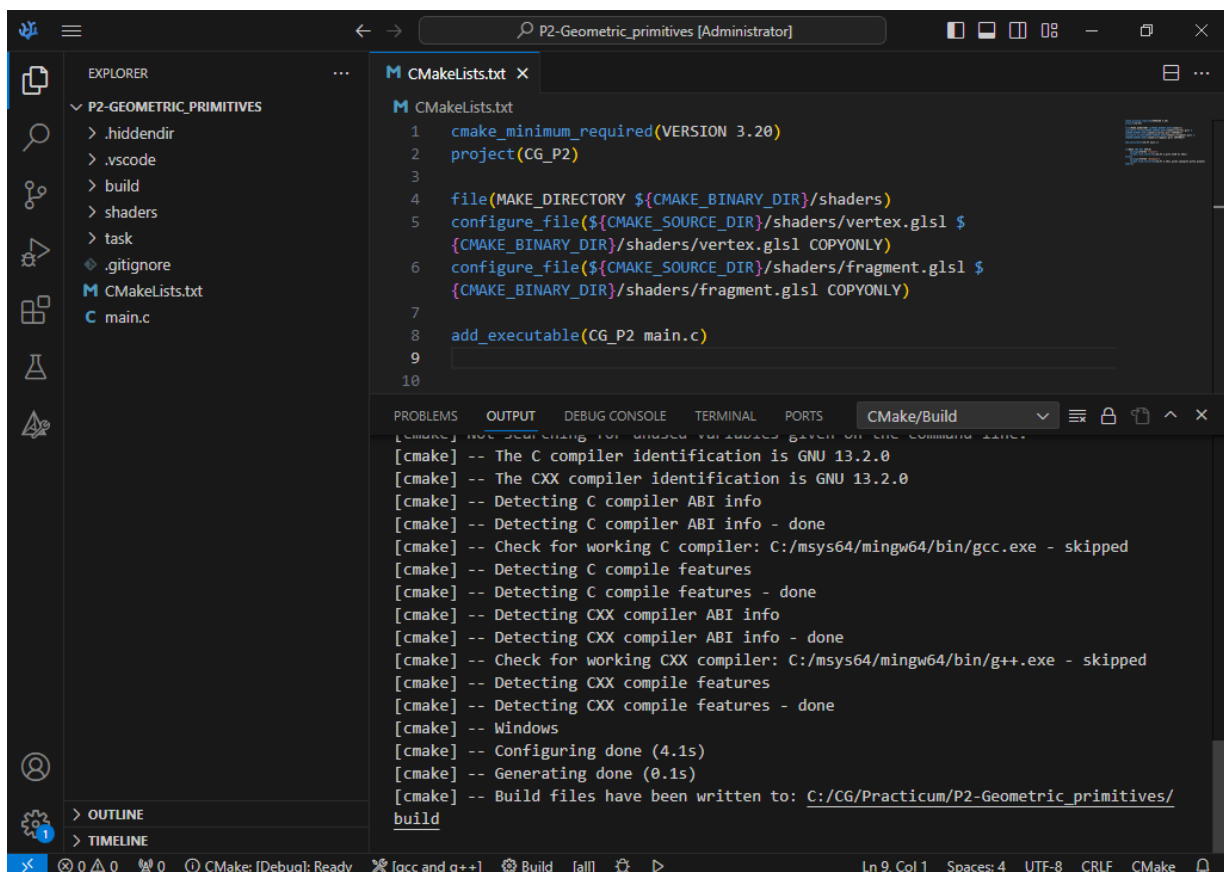


Рис. 8: Возможные варианты путей к компиляторам для CMake



Теперь можно компилировать и запустить собранное приложение **либо** с помощью комбинаций клавиш Ctrl+Shift+P и прописыванием команд, например, ">CMake: Build", ">CMake: Debug" и т.д., **либо** с помощью интерфейса небольшой панели (синей), расположенной в нижней части ИСР VSCodium.