## Установка компилятора в Windows

Мы используем компилятор языка С++ от MinGW g++.exe. Cначала компилятор нужно установить в компьютер. Для установки MinGW для работы с VS Code в Windows можно следовать таким рекомендациям:

1. Загрузите установщик MinGW. Для этого перейдите на сайт mingw.org и нажмите на ссылку «Download/Installer».
2. Установите MinGW и дождитесь появления «Диспетчера установки MinGW». В нём выберите mingw32-gcc-g++ (лучше выбрать все пункты), затем отметьте «Отметить для установки».
3. В меню в верхнем левом углу выберите «Установка > Применить изменения». Дождитесь полной установки. Во время этого процесса убедитесь в наличии стабильного подключения к интернету.
4. Добавьте каталог MinGW в переменную среды PATH. После установки MinGW может находиться по пути C:\MinGW\bin. Добавьте этот каталог в переменную PATH.
5. После установки также необходимо установить расширения в VS Code, например, C/C++, C/C++ Extension Pack, C/C++ Compile Run и Code Runner.

Чтобы проверить, правильно ли установлены и доступны ли ваши инструменты MinGW-w64, откройте терминал и введите командную строку:

gcc --version

g++ --version

gdb –version

Для каждой команды вы должны увидеть вывод на экран сообщения, в котором указано, какие версии GCC, g++ и GDB у вас установлены. Если в терминале Windows запускаются указанные инструменты, значит компилятор установлен в операционную систему и Visual Studio Code сможет предложить к выбору установленный компилятор при запуске программы.

## Подключение библиотеки WinSocket2 к компоновщику

Убедитесь, что среда сборки связана с файлом библиотеки Winsock Ws2\_32.lib. Приложения, использующие Winsock, должны быть связаны с файлом библиотеки Ws2\_32.lib. Информация компоновщику передаётся прямо в тексте программы \*.cpp. Комментарий #pragma указывает компоновщику, что необходим файл *Ws2\_32.lib.*

Используйте Winsock API, включив заголовочные файлы Winsock 2. Заголовочный файл «*Winsock2.h»* содержит большинство функций, структур и определений Winsock. Заголовочный файл «*Ws2tcpip.h»* содержит определения, представленные в документе WinSock 2 Protocol-Specific Annex для TCP/IP, который включает новые функции и структуры, используемые для получения IP-адресов.

Текст программы выглядит так:

#include <winsock2.h>

#include <ws2tcpip.h>

#include <stdio.h>

#pragma comment(lib, "Ws2\_32.lib")

int main() {

return 0;

}

## Подключение библиотеки WinSocket2 к компилятору

Компилятору тоже нужно указать, чтобы он подключил к использованию библиотеку «Ws2\_32.lib». Делается это с использованием ключа «-l» в аргументах строки запуска компилятора. Мы используем компилятор языка С++ от MinGW «g++.exe», он при первом запуске создаёт для себя в папке рабочей области подпапку «.vscode» со стандартным файлом «tasks.json», в котором прописывает параметры для запуска. Ключ нужно добавить к аргументам. У нас файл «tasks.json» выглядит так:

{

    "tasks": [

        {

            "type": "cppbuild",

            "label": "C/C++: g++.exe сборка активного файла",

            "command": "D:\\MinGW\\bin\\g++.exe",

            "args": [

                "-fdiagnostics-color=always",

                "-g",

                "${file}",

                "-o",

                "${fileDirname}\\${fileBasenameNoExtension}.exe",

                "-lWs2\_32"

            ],

            "options": {

                "cwd": "${fileDirname}"

            },

            "problemMatcher": [

                "$gcc"

            ],

            "group": {

                "kind": "build",

                "isDefault": true

            },

            "detail": "Задача создана отладчиком."

        },

     ],

    "version": "2.0.0"

}

Красным цветом выделена добавленная строка с ключом подключения библиотеки.

ВАЖНО, чтобы большие и маленькие буквы в имени файла соответствовали имени реального файла! Стандартное имя файла библиотеки именно такое: «Ws2\_32».