

Как добавить поддержку [Google protobuf](#) в MSYS2 (MinGW)

Поскольку я пишу код на C/C++ для Microsoft Windows с использованием компиляторов GNU (GCC) и IDE [Visual Studio Code](#), то:

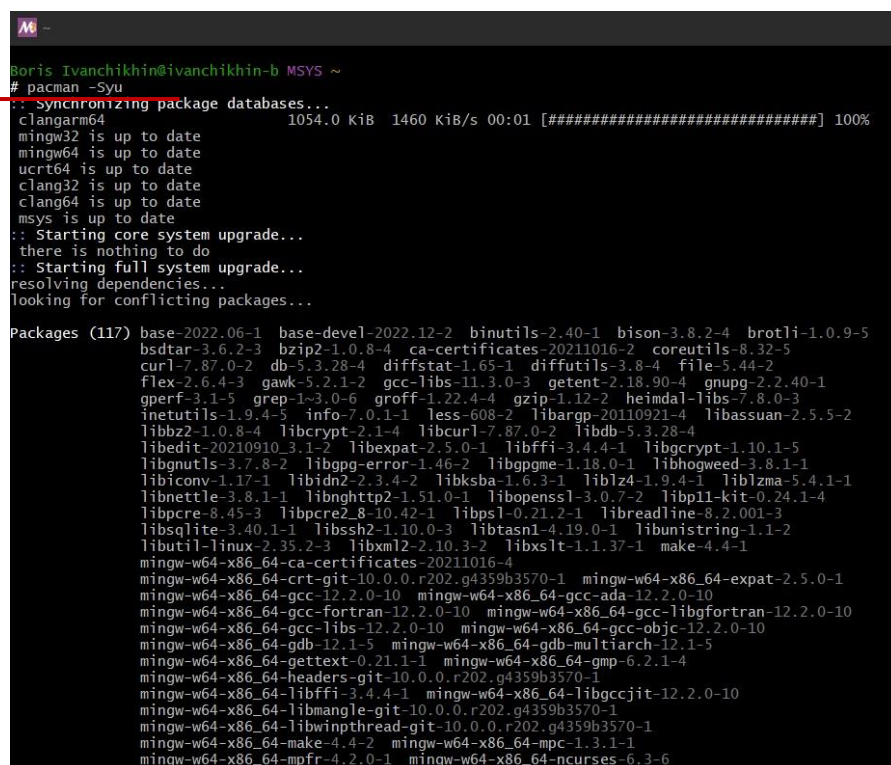
- данная короткая инструкция поясняет каким образом добавить поддержку Protocol Buffers (Google's data interchange format) в указанный инструментарий разработки.

В качестве дистрибутива для всего набора инструментов и библиотек, обеспечивающих среду для создания, установки и запуска собственного программного обеспечения установлен [MSYS2](#). Дистрибутив состоит из терминала командной строки [mintty](#), `bash`, систем контроля версий, таких как `git` и `subversion`. Инструменты: `tar` и `awk`, и даже систем сборки, такие как `autotools`. Все они основаны на модифицированной версии [Cygwin](#). Несмотря на то, что некоторые из частей основаны на `Cygwin`, основное внимание в `MSYS2` уделяется обеспечению среды сборки для собственного программного обеспечения Windows. А части, использующие `Cygwin`, сведены к минимуму. `MSYS2` предоставляет актуальные нативные сборки для GCC, `mingw-w64`, `CPython`, `CMake`, `Meson`, `OpenSSL`, `FFmpeg`, `Rust`, `Ruby` и других.

Чтобы обеспечить простую установку пакетов и способ их обновления, в `MSYS2` есть система управления пакетами под названием [Pacman](#), которая должна быть знакома пользователям Arch Linux. `Pacman` предоставляет множество функций, таких как разрешение зависимостей и простые обновления системы. Репозиторий содержит [более 2600 готовых к установке пакетов](#).

Для того, чтобы добавить поддержку `Google protobuf` в `MSYS2 (MinGW)` нужно запустить "`MSYS2 MSYS`" (`bash`) из меню "Пуск". И далее обновить базовые пакеты с помощью команд:

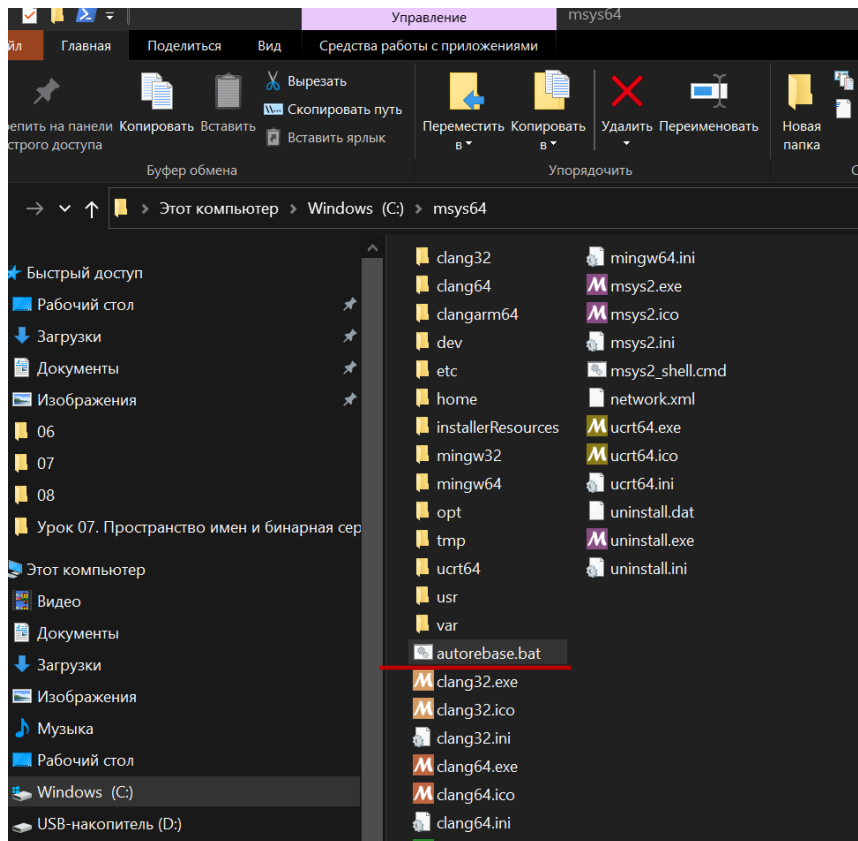
- `pacman -Syu`



```
Boris Ivanchikhin@ivanchikhin-b MSYS ~  
# pacman -Syu  
:: Synchronizing package databases...  
clangarm64 1054.0 KiB 1460 KiB/s 00:01 [#####] 100%  
mingw32 is up to date  
mingw64 is up to date  
ucrt64 is up to date  
clang32 is up to date  
clang64 is up to date  
msys is up to date  
:: Starting core system upgrade...  
there is nothing to do  
:: Starting full system upgrade...  
resolving dependencies...  
looking for conflicting packages...  
  
Packages (117) base-2022.06-1 base-devel-2022.12-2 binutils-2.40-1 bison-3.8.2-4 brotli-1.0.9-5  
bsdtar-3.6.2-3 bzip2-1.0.8-4 ca-certificates-20211016-2 coreutils-8.32-5  
curl-7.87.0-2 db-5.3.28-4 diffstat-1.65-1 diffutils-3.8-4 file-5.44-2  
flex-2.6.4-3 gawk-5.2.1-2 gcc-libs-11.3.0-3 getent-2.18.90-4 gnupg-2.2.40-1  
gperf-3.1-5 grep-3.0-6 groff-1.22.4-4 gzip-1.12-2 heimdal-libs-7.8.0-3  
inetutils-1.9.4-5 info-7.0.1-1 less-608-2 libargp-20110921-4 libassuan-2.5.5-2  
libbz2-1.0.8-4 libcrypto-2.1.4 libcurl-7.87.0-2 libdb-5.3.28-4  
libedit-20210910.3.1-2 libexpat-2.5.0-1 libffi-3.4.4-1 libgcrypt-1.10.1-5  
libgnutls-3.7.8-2 libgpg-error-1.46-2 libgpgme-1.18.0-1 libhogweed-3.8.1-1  
libiconv-1.17-1 libidn2-2.3.4-2 libksba-1.6.3-1 liblz4-1.9.4-1 liblzma-5.4.1-1  
libnettle-3.8.1-1 libnghttp2-1.51.0-1 libopenssl-3.0.7-2 libp11-kit-0.24.1-4  
libpcre-8.45-3 libpcre2-10.42-1 libpsl-0.21.2-1 libreadline-8.2.001-3  
libsqlite-3.40.1-1 libssh2-1.10.0-3 libtasn1-4.19.0-1 libunistring-1.1-2  
libutil-linux-2.35.2-3 libxml2-2.10.3-2 libxslt-1.1.37-1 make-4.4-1  
mingw-w64-x86_64-ca-certificates-20211016-4  
mingw-w64-x86_64-crt-git-10.0.0.r202.g4359b3570-1 mingw-w64-x86_64-expat-2.5.0-1  
mingw-w64-x86_64-gcc-12.2.0-10 mingw-w64-x86_64-gcc-ada-12.2.0-10  
mingw-w64-x86_64-gcc-fortran-12.2.0-10 mingw-w64-x86_64-gcc-libgfortran-12.2.0-10  
mingw-w64-x86_64-gcc-libs-12.2.0-10 mingw-w64-x86_64-gcc-objc-12.2.0-10  
mingw-w64-x86_64-gdb-12.1-5 mingw-w64-x86_64-gdb-multiarch-12.1-5  
mingw-w64-x86_64-gettext-0.21.1-1 mingw-w64-x86_64-gmp-6.2.1-4  
mingw-w64-x86_64-headers-git-10.0.0.r202.g4359b3570-1  
mingw-w64-x86_64-libffi-3.4.4-1 mingw-w64-x86_64-libgccjit-12.2.0-10  
mingw-w64-x86_64-libmangle-git-10.0.0.r202.g4359b3570-1  
mingw-w64-x86_64-libwinpthread-git-10.0.0.r202.g4359b3570-1  
mingw-w64-x86_64-make-4.4-2 mingw-w64-x86_64-mpc-1.3.1-1  
mingw-w64-x86_64-mpfr-4.2.0-1 mingw-w64-x86_64-ncurses-6.3-6
```

- `pacman -S --needed base-devel mingw-w64-x86_64-toolchain`

- после этого закрыть все оболочки MSYS2 и запустить `autorebase.bat` в корневом каталоге MSYS2:



- теперь, собственно, уже можно установить поддержку [Google protobuf](https://github.com/google/protobuf) в MSYS2(MinGW):

`pacman -S mingw-w64-x86_64-protobuf`

```
Boris Ivanchikhin@ivanchikhin-b MSYS ~
# pacman -S mingw-w64-x86_64-protobuf
resolving dependencies...
looking for conflicting packages...

Packages (1) mingw-w64-x86_64-protobuf-21.12-1

Total Download Size:    3.04 MiB
Total Installed Size:  30.84 MiB

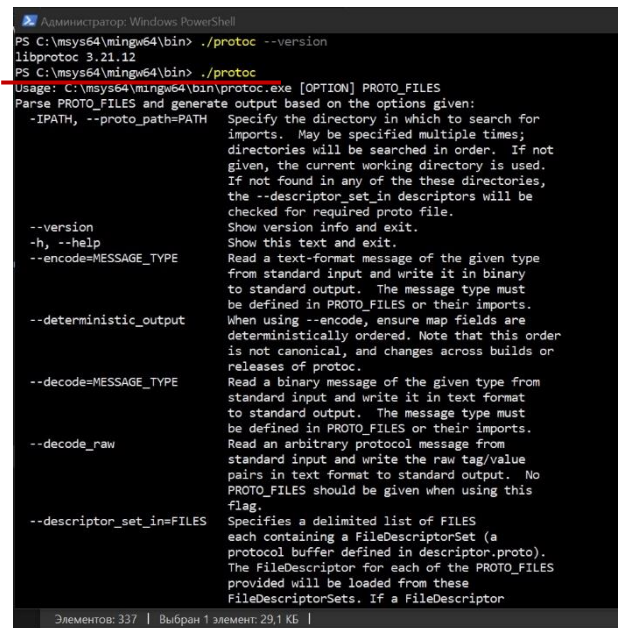
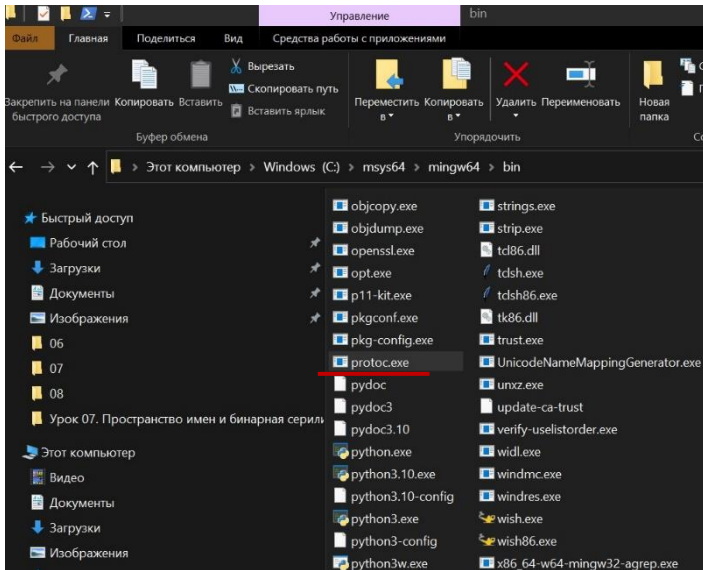
:: Proceed with installation? [Y/n] y
:: Retrieving packages...
 mingw-w64-x86_64-protobuf-21.12-1-any           3.0 MiB   2.40 MiB/s   00:01 [#####] 100%
(1/1) checking keys in keyring                    [#####] 100%
(1/1) checking package integrity                  [#####] 100%
(1/1) loading package files                      [#####] 100%
(1/1) checking for file conflicts                 [#####] 100%
(1/1) checking available disk space              [#####] 100%
:: Processing package changes...
(1/1) installing mingw-w64-x86_64-protobuf        [#####] 100%

Boris Ivanchikhin@ivanchikhin-b MSYS ~
#
```

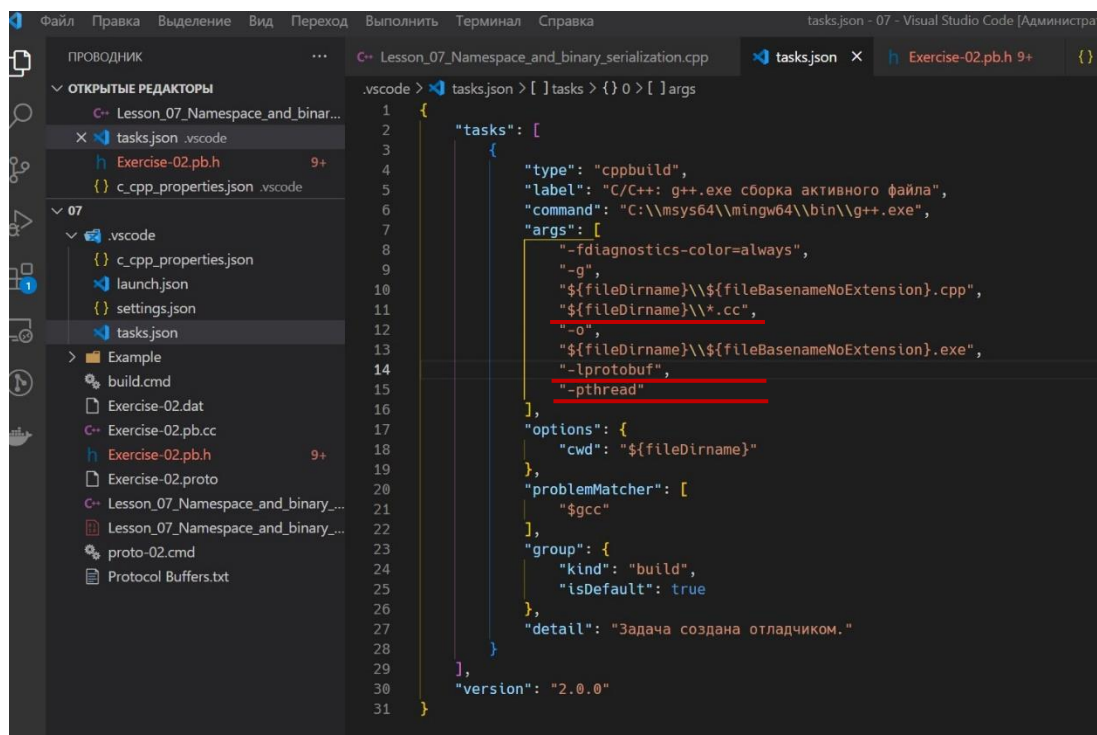
В дальнейшем, для успешной сборки проектов с использованием Google protobuf в MSYS2 (MinGW) компилятору g++ потребуется указание флагов:

- `-lprotobuf`
- `-pthread`

Теперь доступна утилита `protoc` в соответствующем каталоге установки (в моем случае это - `C:\msys64\mingw64\bin`):



Соответственно, для сборки проектов в среде Visual Studio Code необходимо внести изменения в файл `tasks.json`. Сделать это можно, например, таким образом:



Ссылки:

[How to build Google's protobuf in Windows using MinGW?](#)

[Разбор Protobuf в Visual Studio под C++](#)

[Руководство по языку \(прото3\)](#)