

PUBLICACIONES 2020

Como instalar GCC en windows 10

¿Que es GCC?

GCC viene de *GNU Compiler Collection* o colección de compiladores de GNU. Pero...

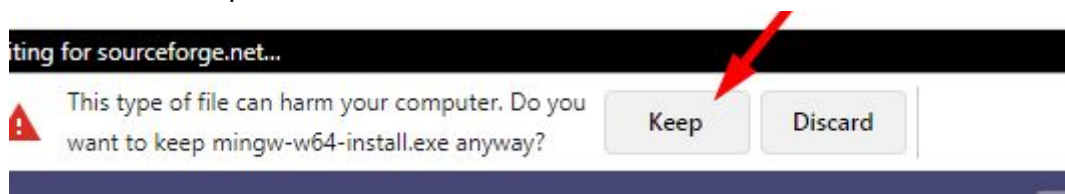
¿Que es un compilador? Es un programa informático que traduce un lenguaje de programación a otro lenguaje, la mayoría de veces a un lenguaje de bajo nivel. Ya, y ¿Que es lenguaje de programación de bajo nivel? Una forma simple de explicarlo sería: es aquel que está más cerca de comunicarse directamente con el hardware y ejemplo es “assembler” o lenguaje ensamblador.

Así el lenguaje C viene siendo más bajo nivel que Python, pero assembler viene siendo más bajo nivel que C y el código binario sería más bajo nivel que assembler. De esto se desprende que mientras más alto el nivel más fácil es de entender y programar y vice versa.

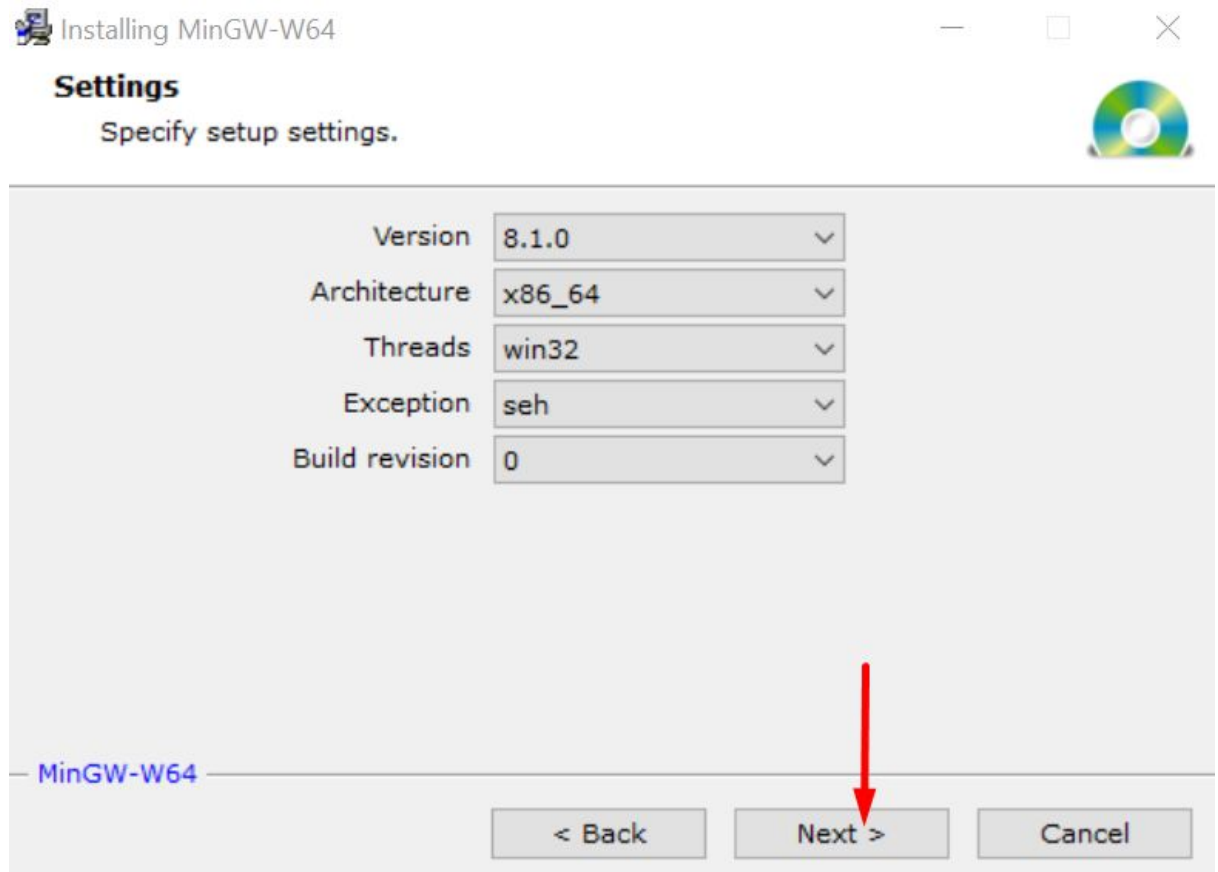
Bueno ¿Y para que sirve? GCC me sirve para compilar programas en C y generar un ejecutable o binario.

¿Como instalar?

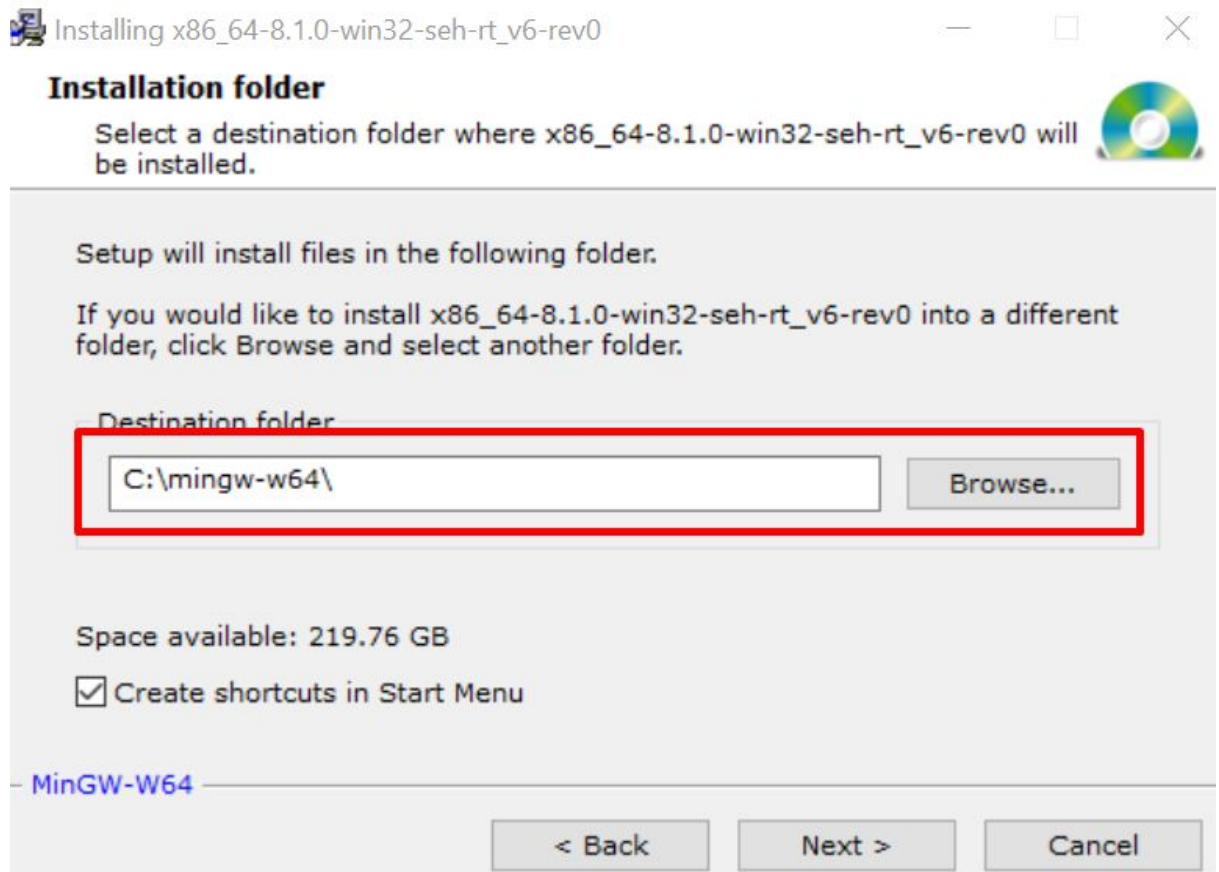
- Descargar directamente MinGW-w64 desde <https://sourceforge.net/projects/mingw-w64/files/Toolchains%20targetting%20Win32/Personal%20Builds/mingw-builds/installer/mingw-w64-install.exe/download>
- Dar click en “Keep”.



- Abrir instalador y dar a "Next". Seleccionar las opciones de la imagen de abajo y darle a "Next".



- Cambiar el destino de la instalación al que se muestra en la imagen.
“C:\mingw-w64\” y darle a “Next”.

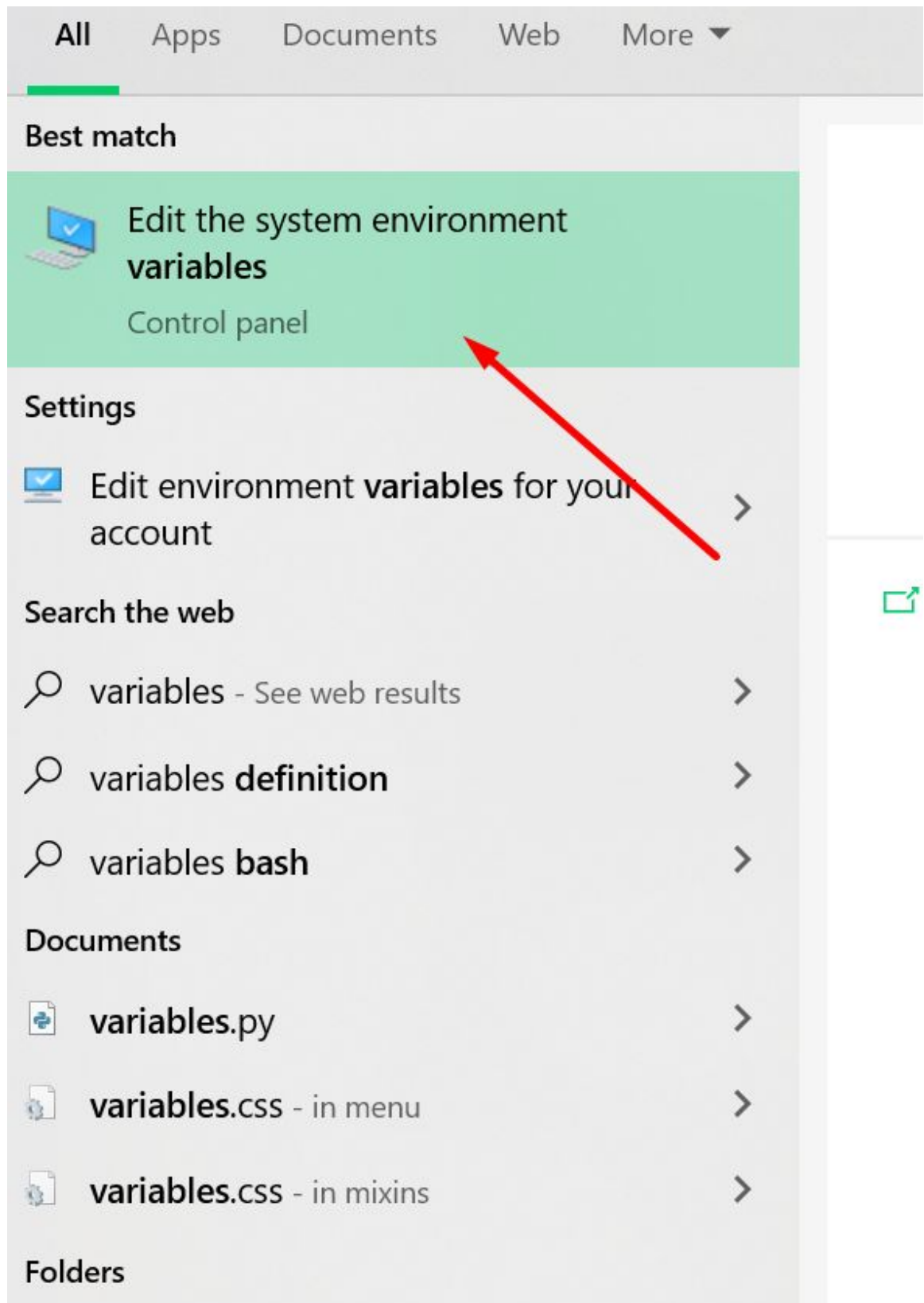


- Esperamos que finalice la instalación y le damos a "Finish".

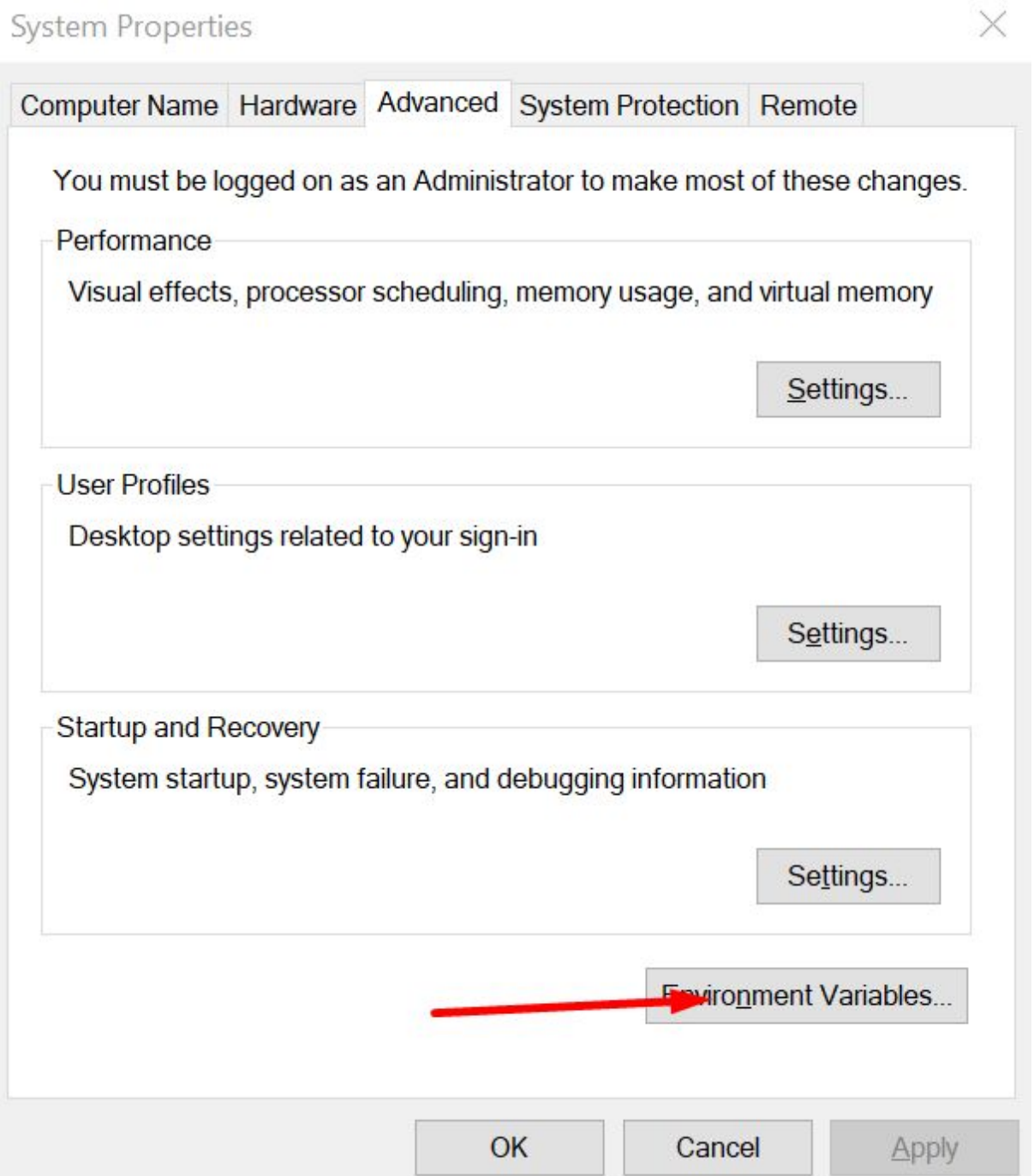


- Una vez que está todo instalado tenemos que agregar las variables de ambiente. Presionamos la tecla de inicio de "Windows" y escribimos "variable", nos sugerirá

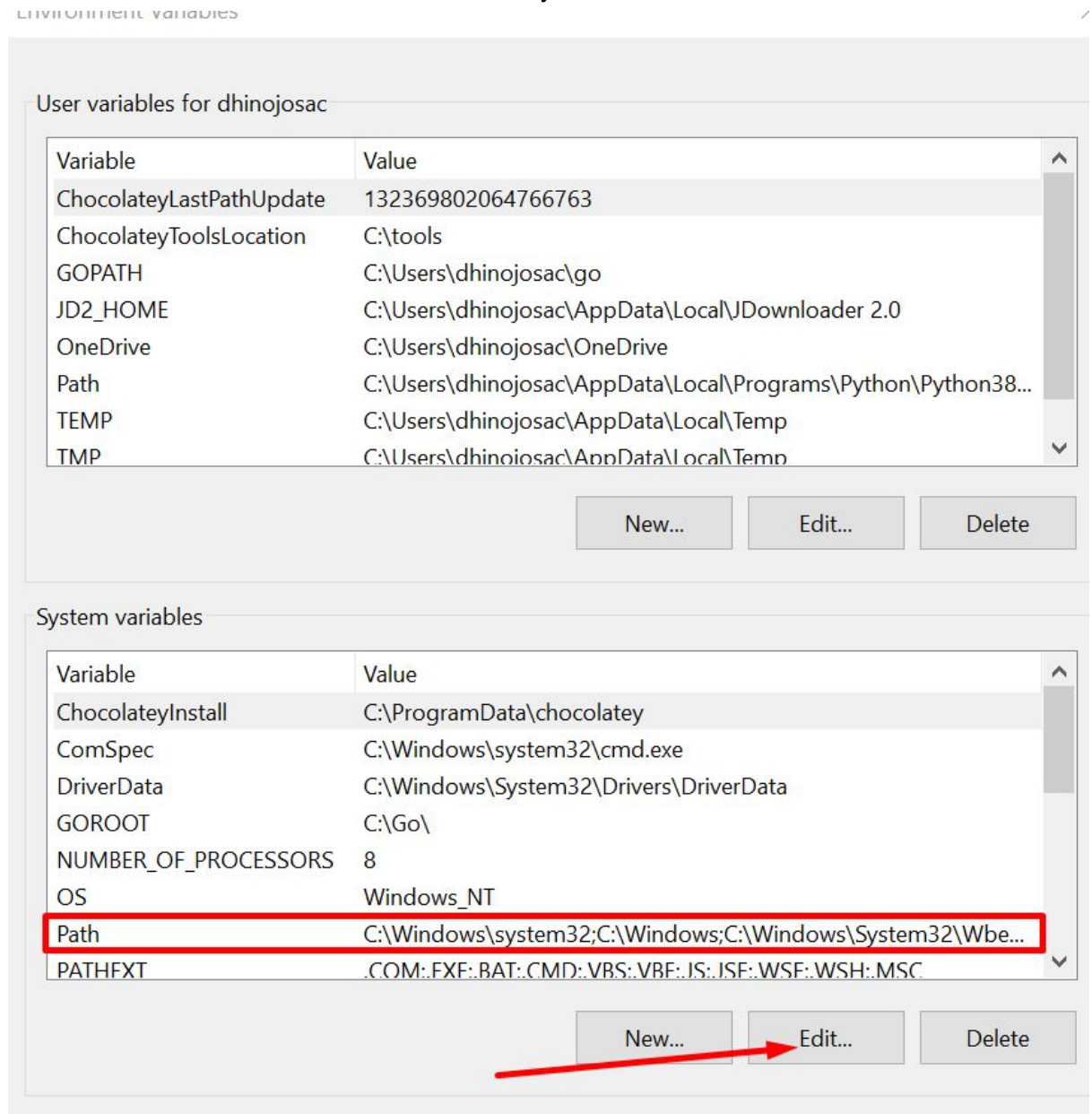
“editar las variables de ambiente del sistema”. Entramos a ella.



- En propiedades del sistema vamos a “Variables de Ambiente...” o “Environment Variables...” (Siempre depende del idioma del sistema operativo)

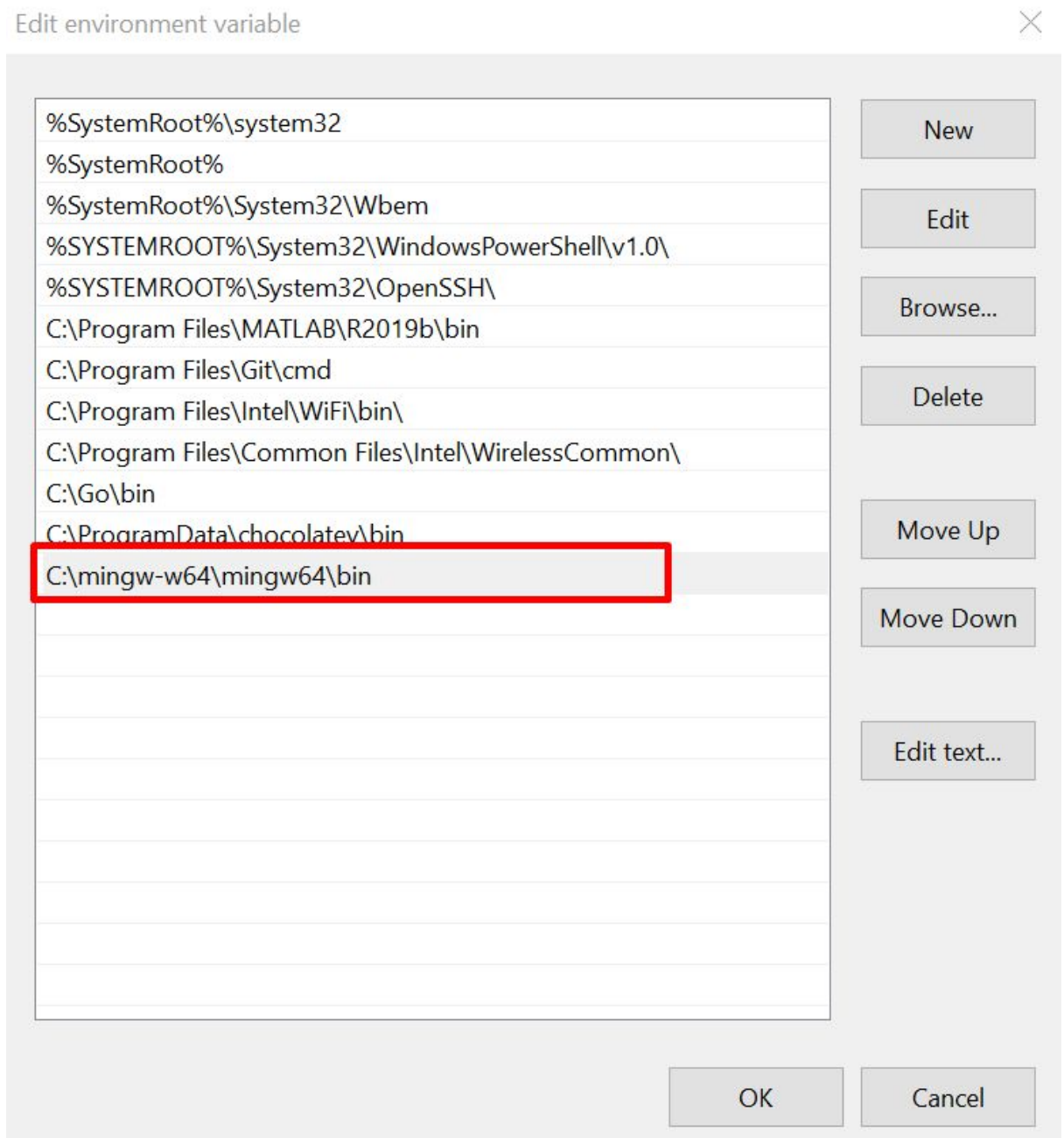


- Buscamos el Path en las variables del sistema y damos click al botón “Edit...”.



- Luego presionamos “New” para agregar nuestra nueva variable. Agregamos la ruta de la carpeta “bin”, donde instalamos MingGW, que en nuestro caso es

“C:\mingw-w64\mingw64\bin” y damos “OK”.



- Ahora abrimos una ventana del interprete de comandos de windows presionando tecla “Windows”+R y escribimos “cmd” y damos a “OK”. Una vez abierta, escribimos

“gcc -v”. Si todo está correcto debería verse como la foto de abajo.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\dhinojosac>gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=C:/mingw-w64/mingw64/bin/./libexec/gcc/x86_64-w64-mingw32/8.1.0/lto-wrapper.exe
Target: x86_64-w64-mingw32
Configured with: ../../src/gcc-8.1.0/configure --host=x86_64-w64-mingw32 --build=x86_64-w64-mingw32 --target=x86_64-w64-mingw32 --prefix=/mingw64 --with-sysroot=/c/mingw810/x86_64-810-win32-seh-rt_v6-rev0/mingw64 --enable-shared --enable-static --disable-multilib --enable-languages=c,c++,fortran,lto --enable-libstdcxx-time=yes --enable-threads=win32 --enable-libgomp --enable-libatomic --enable-lto --enable-graphite --enable-checking=release --enable-fully-dynamic-string --enable-version-specific-runtime-libs --disable-libstdcxx-pch --disable-libstdcxx-debug --enable-bootstrap --disable-rpath --disable-win32-registry --disable-nls --disable-werror --disable-symvers --with-gnu-as --with-gnu-ld --with-arch=nocore --with-tune=core2 --with-libiconv --with-system-zlib --with-gmp=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-mpfr=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-mpc=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-isl=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-pkgversion='x86_64-win32-seh-rt_v6-rev0, Built by MinGW-W64 project' --with-bugurl=https://sourceforge.net/projects/mingw-w64 CFLAGS='-O2 -pipe -fno-ident -I/c/mingw810/x86_64-810-win32-seh-rt_v6-rev0/mingw64/opt/include -I/c/mingw810/prerequisites/x86_64-zlib-static/include -I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' CXXFLAGS='-O2 -pipe -fno-ident -I/c/mingw810/x86_64-810-win32-seh-rt_v6-rev0/mingw64/opt/include -I/c/mingw810/prerequisites/x86_64-zlib-static/include -I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' CPPFLAGS='-I/c/mingw810/x86_64-810-win32-seh-rt_v6-rev0/mingw64/opt/include -I/c/mingw810/prerequisites/x86_64-zlib-static/include -I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' LDFLAGS='-pipe -fno-ident -L/c/mingw810/x86_64-810-win32-seh-rt_v6-rev0/mingw64/opt/lib -L/c/mingw810/prerequisites/x86_64-zlib-static/lib -L/c/mingw810/prerequisites/x86_64-w64-mingw32-static/lib'
Thread model: win32
gcc version 8.1.0 (x86_64-win32-seh-rt_v6-rev0, Built by MinGW-W64 project)

C:\Users\dhinojosac>
```

- Si no se teaste bien la variable de entorno deberia verse de esta forma.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.900]
(c) 2019 Microsoft Corporation. All rights reserved.

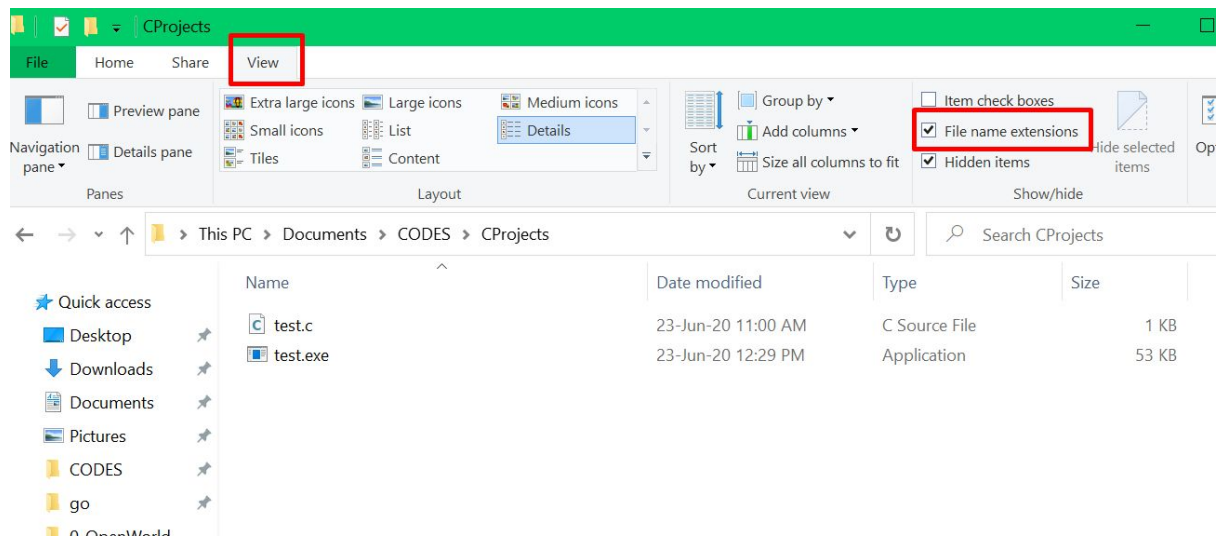
C:\Users\dhinojosac>gcc -v
'gcc' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\dhinojosac>
```

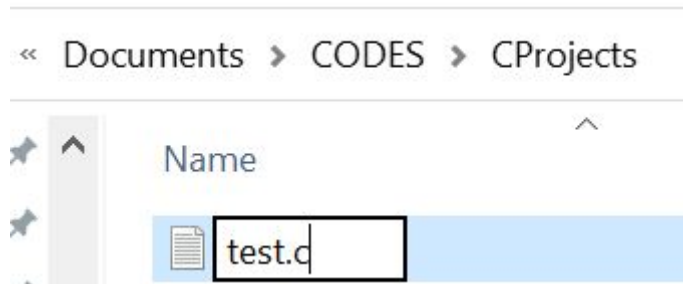
A probar gcc...

- Abrimos una carpeta donde queramos crear nuestro primer código en C. En mi caso la carpeta “CProjects”.

- A veces está configurado para no mostrar las extensiones de los archivos, pero la puedes activar como en la primera imagen de abajo.



- Damos botón derecho crear “nuevo”> “Documento de texto”. Y nombramos el archivo “test.c”, asegurarse de borrar la extensión .txt.



- Abrimos nuestro archivo y copiamos el siguiente ejemplo en C.

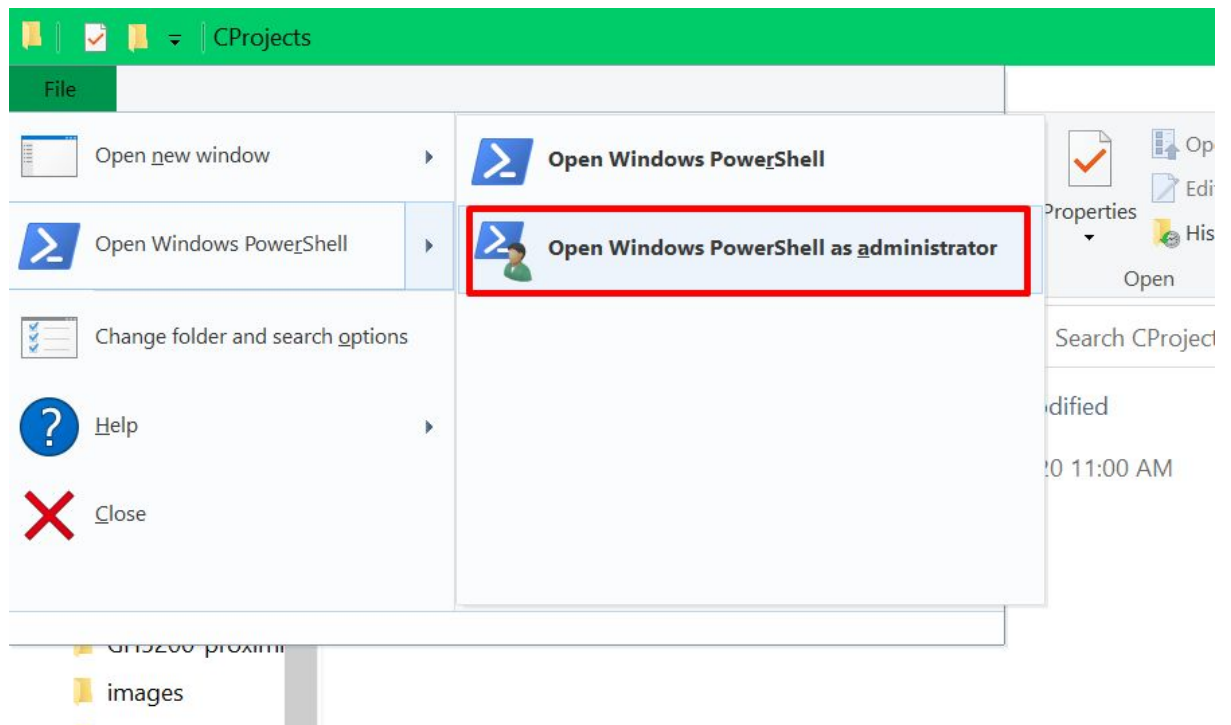
```
#include <stdio.h>

int main(){
    printf("Mi primer codigo compilado en GCC.\n");
    printf("dhinojosac 2020\n");

    return 0;
}
```

- Luego abrimos una consola como administrador. En la carpeta dar a “Archivo”> “Abrir Windows PowerShell”> “Abrir Windows PowerShell como

administrador” y damos a “Si”.



- Se nos abrirá una pantalla azul, en la que introducimos los comandos básicos para compilar nuestro programa. “gcc test.c -o test” y damos enter. Si todo salió bien no nos mostrará ningún mensaje.

```
Administrator: Windows PowerShell
PS C:\Users\dhinojosac\Documents\CODES\CProjects> gcc .\test.c -o test.exe
PS C:\Users\dhinojosac\Documents\CODES\CProjects>
```

- Si todo sale bien al escribir la línea de comando “dir” veremos el nuevo archivo test.exe.

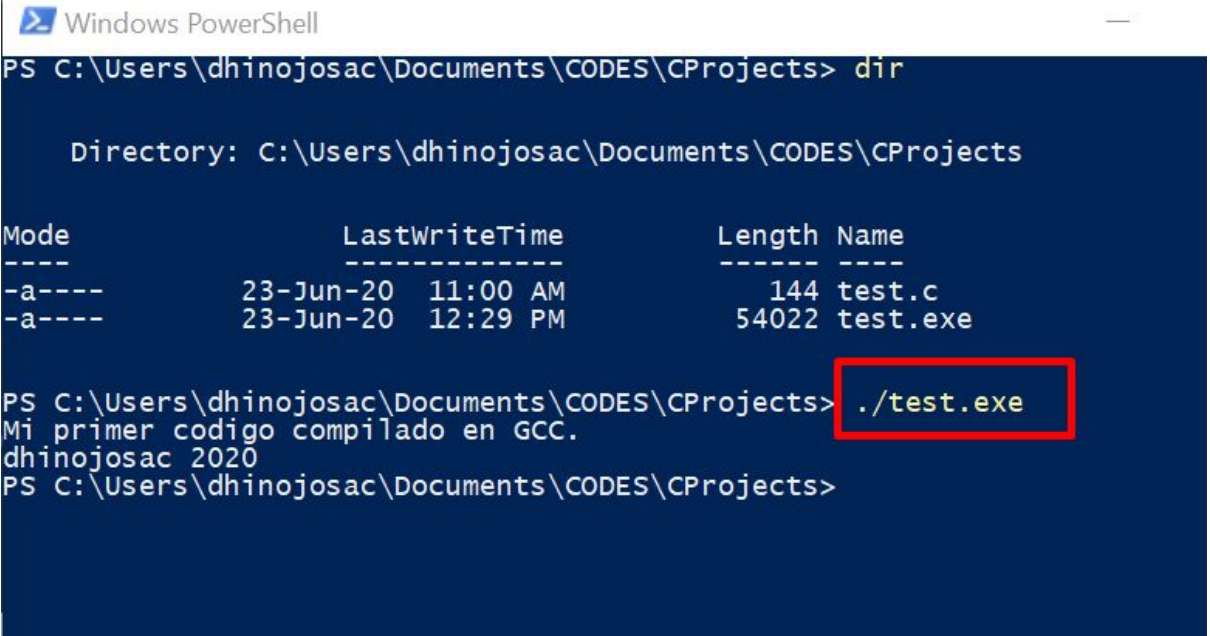
```
Windows PowerShell
PS C:\Users\dhinojosac\Documents\CODES\CProjects> dir

Directory: C:\Users\dhinojosac\Documents\CODES\CProjects

Mode                LastWriteTime         Length Name
----                -
-a----             23-Jun-20   11:00 AM          144 test.c
-a----             23-Jun-20   12:29 PM       54022 test.exe

PS C:\Users\dhinojosac\Documents\CODES\CProjects>
```

- Para probar nuestro primer código compilado, lo ejecutamos escribiendo en la línea de comandos `./test.exe`.



A screenshot of a Windows PowerShell terminal window. The title bar at the top reads "Windows PowerShell". The terminal shows the following sequence of commands and output:

```
PS C:\Users\dhinojosac\Documents\CODES\CProjects> dir

Directory: C:\Users\dhinojosac\Documents\CODES\CProjects

Mode                LastWriteTime         Length Name
----                -
-a----            23-Jun-20  11:00 AM           144 test.c
-a----            23-Jun-20  12:29 PM        54022 test.exe

PS C:\Users\dhinojosac\Documents\CODES\CProjects> ./test.exe
Mi primer código compilado en GCC.
dhinojosac 2020
PS C:\Users\dhinojosac\Documents\CODES\CProjects>
```

The command `./test.exe` is highlighted with a red rectangular box.