

INTRODUCCIÓN AL LENGUAJE C / C++



Aprenda desde la Consola de Comandos

Oscar Núñez Mori

INDICE

- Aspectos Legales
- ¿Porque C y C++?
- Breve Introducción
- Instalación de Software
- El PATH de Windows
- Terminal CMD
- Comandos DOS
- Terminal URCT64
- Comandos MINGW64
- Hola Mundo

ASPECTOS LEGALES

Este curso fue desarrollada con la asistencia de [ChatGPT](#) (OpenAI), utilizado para estructurar ideas, generar texto técnico y apoyar en la programación. Se estima que más del 50% del contenido fue generado con su ayuda. La validación y edición final fueron realizadas por el autor.

OpenAI (2025a)

En este curso, con fines académicos, se usó el Sistema Windows 10 Pro descargado desde el [Sitio Oficial de Microsoft](#) y se instaló en [VirtualBox](#) corriendo en un sistema [Linux Ubuntu](#) como host.

OpenAI (2025c)

¿Porqué C y C++?

Kernel de Linux

torvalds/**linux**

Linux kernel source tree



16k

Contributors



1

Used by



194k

Stars



56k

Forks



● C 98.3% ● Assembly 0.7% ● Shell 0.4% ● Python 0.2% ● Makefile 0.2% ● Perl 0.1%
● Other 0.1%

El Kernel del Sistema Operativo de Linux está en Lenguaje C

Torvalds (s.f.)

¿Porqué C y C++?

Aviónica

El Lockheed Martin F-35 Joint Strike Fighter (JSF) es una aeronave novedosa por diversas razones, entre las que destaca su software. Según se informa, este **caza de quinta generación consta de más de 10 millones de líneas de código, segmentadas en bloques y escritas principalmente en C y C++**; sin embargo, también utiliza código en el lenguaje de programación Ada del caza militar Lockheed Martin/Boeing F-22 Raptor.



¿Porqué C y C++?

Comunicaciones Industriales



libmodbus

La biblioteca de código abierto más popular para la comunicación con dispositivos Modbus. 🚀

Estado de compilación

Compatibilidad con RTU (serie) y TCP (Ethernet)
Disponible para Linux (paquete), FreeBSD, Mac OS y Windows

Escrito en C

Excelente cobertura de pruebas y documentación

Auditorías de seguridad

Sin dependencias

libmodbus Project. (s.f.)



Desarrollado en C99 con complementos específicos para la arquitectura.

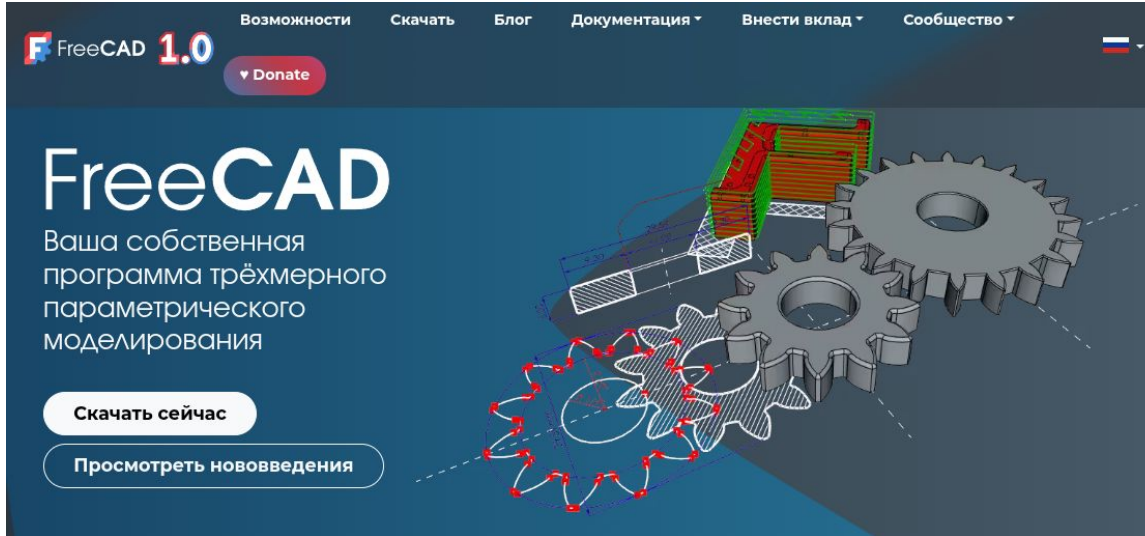
Compatible con Windows, Linux, VxWorks, QNX, Android y diversos sistemas integrados.
Requiere poco espacio de memoria.

open62541 Project. (s.f.)

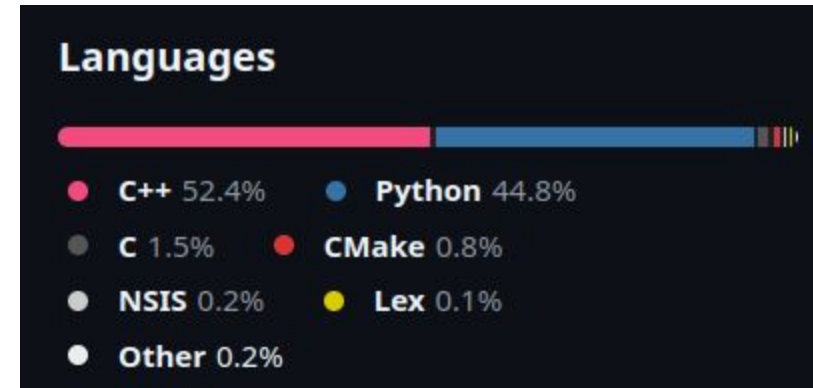
Open62541 es una biblioteca de código abierto que permite implementar el protocolo de comunicación **OPC UA**

¿Porqué C y C++?

Software de Diseño Mecánico



FreeCAD Project. (s.f.)



FreeCAD Project. (s.f.a.)

¿Porqué C y C++?

C++ Game Engine



We make the engine. You make it Unreal.

Unreal Engine is built by developers, for developers, with fair terms for all.
Because everyone with an idea deserves to use the world's most open,
advanced real-time creation tool to bring it to life.

[Get Unreal Engine](#)

[Learn about licensing](#)

Beautiful Light
Deep Worlds SA



Licencia: Gratis hasta que generes más de 1 millón USD (luego 5% de regalías)



- **Especialistas en macrodatos**
- **Ingenieros en tecnofinanzas**
- **Especialistas en IA y aprendizaje automático**
- **Desarrolladores de software y aplicaciones**
- **Especialistas en gestión de seguridad**
- **Especialistas en almacenamiento de datos**
- **Especialistas en vehículos autónomos y eléctricos**

Fuente: Foro Económico Mundial, 2025



- **Diseñadores de UI y UX**
- **Conductores de camiones ligeros o de entrega a domicilio**
- **Especialistas en Internet de las cosas**

Fuente: Foro Económico Mundial, 2025

Historia del Lenguaje C y C++

La historia nos remonta a la figura de **Dennis Ritchie**, quien introdujo por primera vez el lenguaje de programación C en los Laboratorios Bell de AT&T en 1972 y se implementó por primera vez en la computadora DEC PDP-11. Dennis Ritchie usó los conceptos de BCPL y B para desarrollar C y agregó la tipificación de datos y algunas otras características poderosas.

Reclut (s.f.)



Digital News Asia (2012)

Después, en 1979, **Bjarne Stroustrup**, también empleado de Bell AT & T, comenzó a trabajar en el lenguaje C con las clases. Tomó prestadas las características deseables de muchos otros lenguajes como Simula, Ada, ML, CLU y ALGOL 68. Por lo tanto, además de las características del lenguaje C, C ++ también incluía clases, verificación de tipo fuerte, argumento de función predeterminada y herencia básica. Hasta 1983, se llamaba C con clases, y en 1983 se llamaba C ++. Durante 1998, un comité conjunto ANSI-ISO publicó la especificación para los estándares de lenguaje C ++

Reclut (s.f.)



Stroustrup (s.f.)

Software a Instalar



7-Zip. (s.f.)

En Path:

C:\Program Files\7-Zip

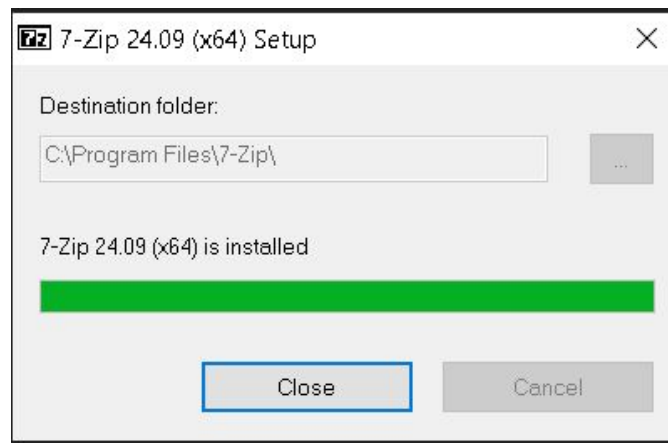
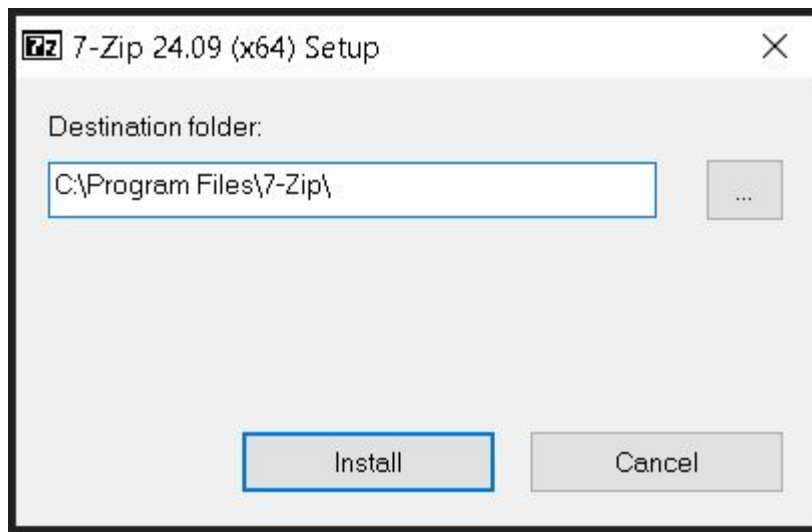
Comprimir en consola

7z a destino.7z origen

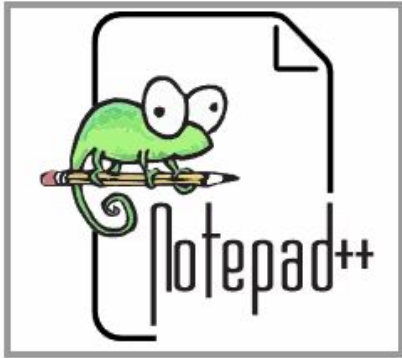
Descomprimir en consola

7z e destino.7z

WinMD5Free



Software a Instalar



`pacman -S mingw-w64-ucrt-x86_64-geany`



`pacman -S mingw-w64-ucrt-x86_64-gnuplot`

COMPILADORES DE C y C++



clang 3.5 is here...

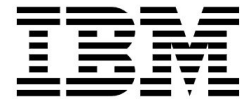


Microsoft C++, C, and Assembler



embarcadero

IBM Open XL C/C++
for AIX



COMPILADOR ELEGIDO



MSYS2 es una herramienta para desarrollar software nativo de Windows, en un entorno similar a Unix para el desarrollo y la compilación de programas en C/C++



CULTURA DE DONACIÓN



MSYS2

COLLECTIVE



Fiscal Host: Open Collective Europe

Open Source Software Distribution and Building Platform for Windows



CONTRIBUTE



BUDGET



ABOUT

SUBMIT EXPENSE

CONTACT



Contribute

Become a financial contributor.

Financial Contributions

RECURRING CONTRIBUTION

Backer

Become a backer for €5.00 per month and support us

STARTS AT

€5 EUR / month

RECURRING CONTRIBUTION

Sponsor

Become a sponsor for €100.00 per month and support us

STARTS AT

€100 EUR / month






CUSTOM CONTRIBUTION

Donation

Make a custom one-time or recurring contribution.

MSYS2 (s.f.)



	Name	Prefix	Toolchain	Architecture	C Library	C++ Library
	MSYS	<code>/usr</code>	gcc	x86_64	cygwin	libstdc++
	UCRT64	<code>/ucrt64</code>	gcc	x86_64	ucrt	libstdc++
	CLANG64	<code>/clang64</code>	llvm	x86_64	ucrt	libc++
	CLANGARM64	<code>/clangarm64</code>	llvm	aarch64	ucrt	libc++
	MINGW64	<code>/mingw64</code>	gcc	x86_64	msvcrt	libstdc++

	UCRT64	<code>/ucrt64</code>	gcc	x86_64	ucrt	libstdc++
---	--------	----------------------	-----	--------	------	-----------

Instalando MSYS2 enfocado a UCRT64

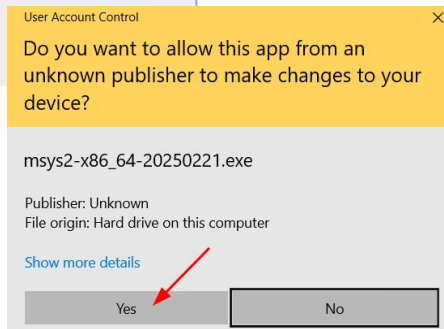
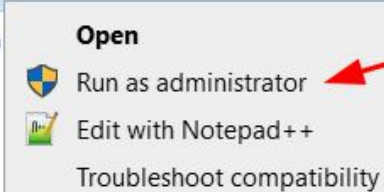


Descargamos en Windows 10 o superior
el archivo **msys2-x86_64-AñoMesDia.exe**

Installation

1. Download the installer: **msys2-x86_64-20250221.exe**

Se escoge la ruta
de instalación:
C:\prg\msys64



MSYS2 Setup

Welcome

Welcome

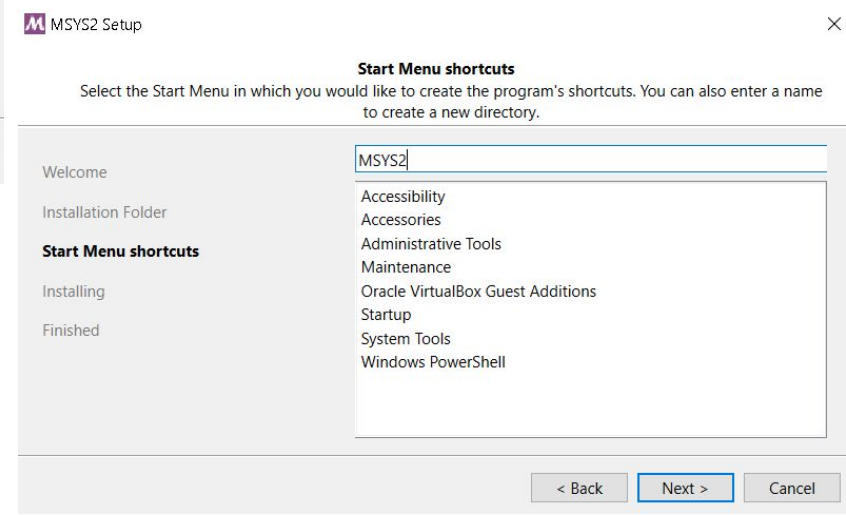
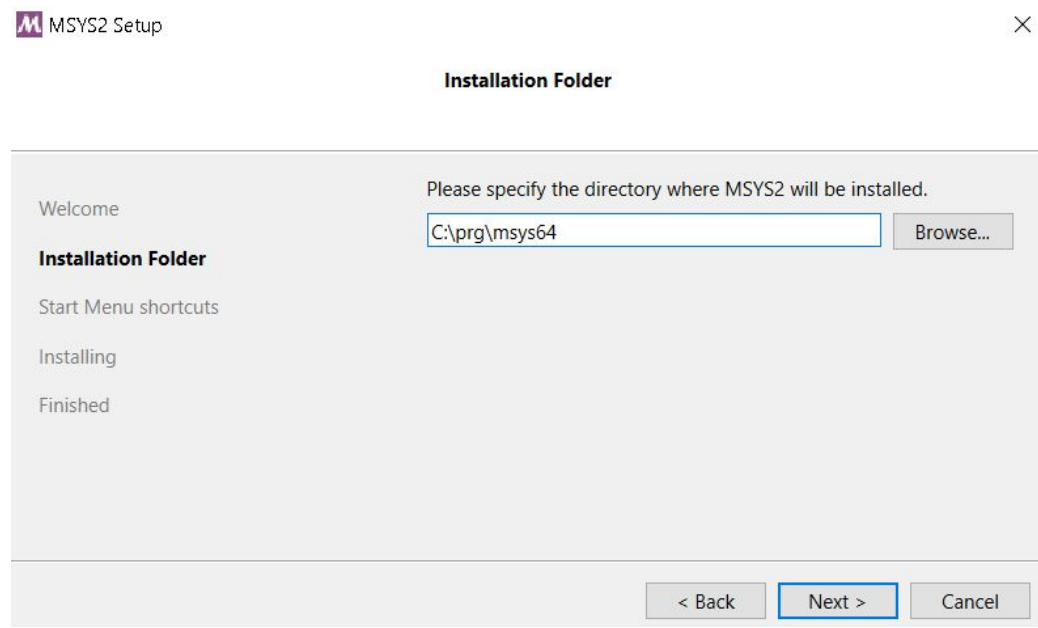
Welcome to the MSYS2 Setup.

Installation Folder
Start Menu shortcuts
Installing
Finished

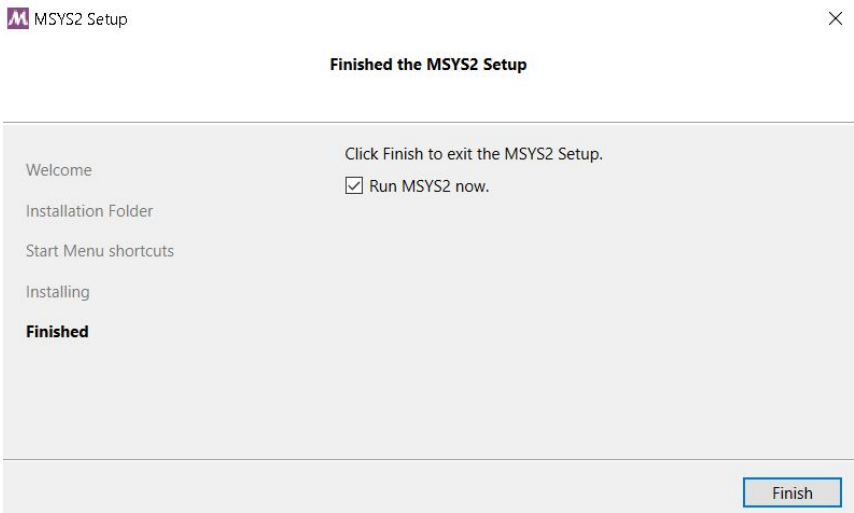
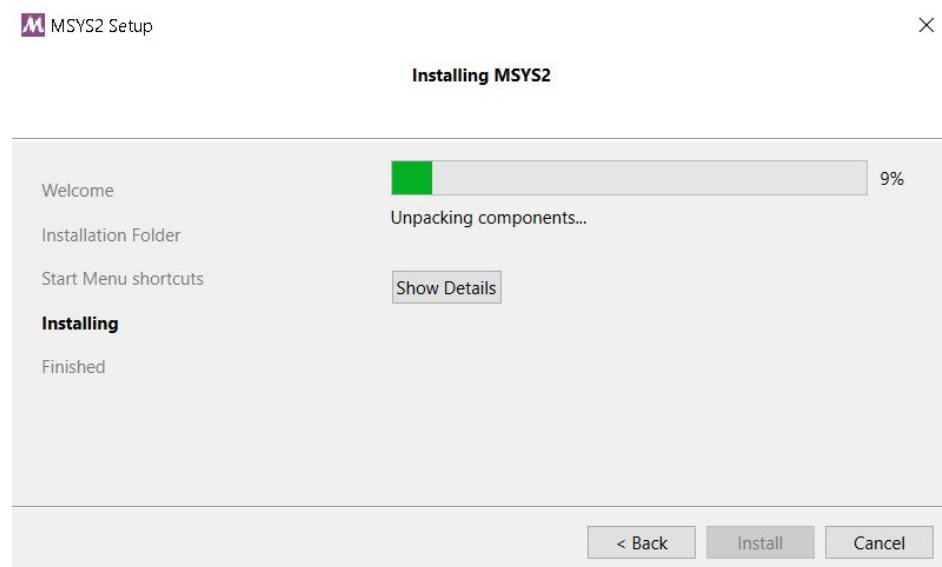
Next >

Quit

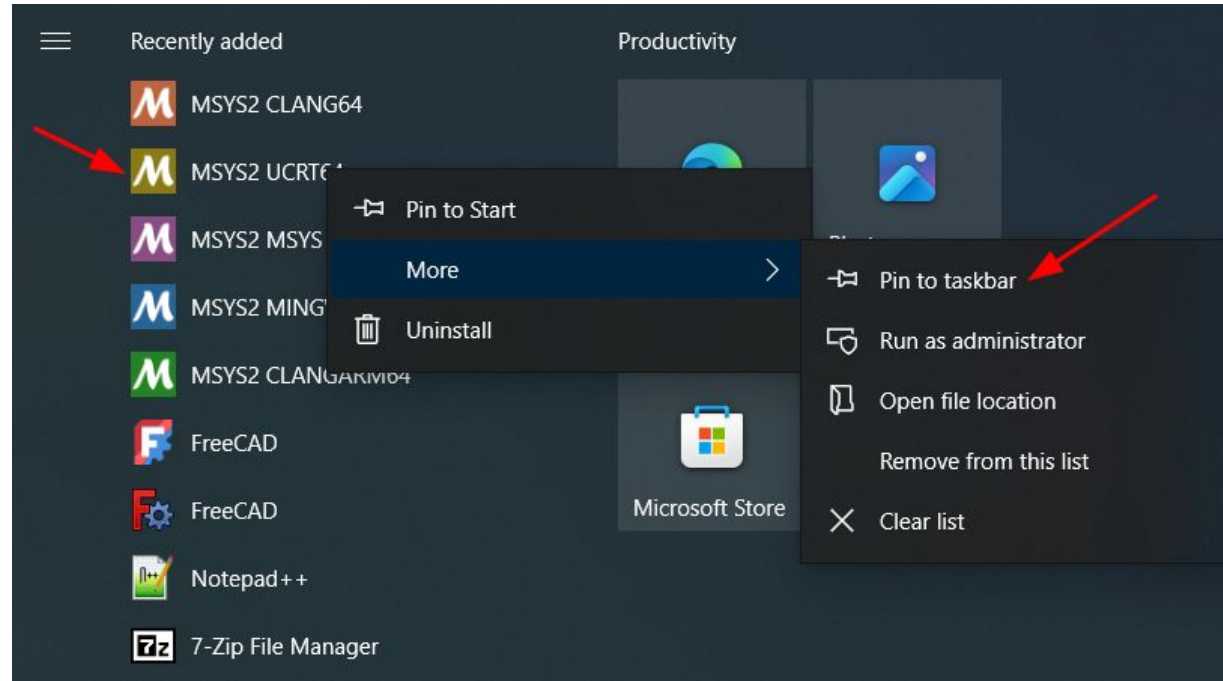
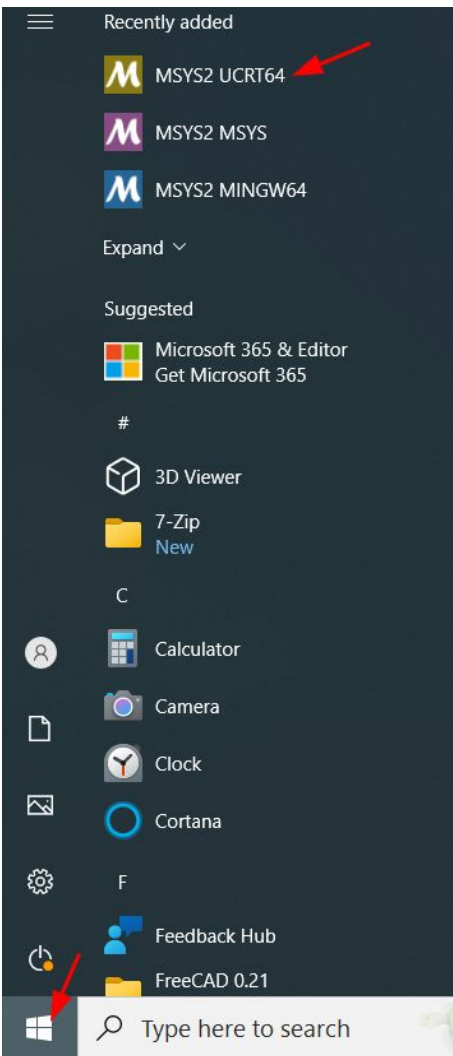
Instalando MSYS2 enfocado a UCRT64



Instalando MSYS2 enfocado a UCRT64



Instalando MSYS2 enfocado a UCRT64



Actualizando los paquetes desde el Terminal o Consola UCRT64



Lanzadera del
Terminal UCRT64

pacman -Syu

```
M ~
oscar@WIN-10 UCRT64 ~
$ pacman -Syu
```

Se repite 3 veces o más
el mismo comando:
pacman -Syu para que
actualice completamente
los paquetes y nos dé algo
así:

```
M ~
mingw32           147.2 KiB   63.2 KiB/s  00:02
mingw64           457.8 KiB   157 KiB/s  00:03
ucrt64            497.2 KiB   181 KiB/s  00:03
clang64           484.5 KiB   159 KiB/s  00:03
msys              373.0 KiB   302 KiB/s  00:01
:: Starting core system upgrade...
warning: terminate other MSYS2 programs before proceeding
resolving dependencies...
looking for conflicting packages...

Packages (3) filesystem-2025.02.23-1
             mintty-1~3.7.8-1
             msys2-runtime-3.6.1-3

Total Download Size: 2.76 MiB
Total Installed Size: 9.00 MiB
Net Upgrade Size:    0.19 MiB

:: Proceed with installation? [Y/n]
```

```
M ~
oscar@WIN-10 UCRT64 ~
$ pacman -Syu
:: Synchronizing package databases...
clangarm64 is up to date
mingw32 is up to date
mingw64 is up to date
ucrt64 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...
there is nothing to do

oscar@WIN-10 UCRT64 ~
$
```

Instalando paquetes con el Comando pacman en el Terminal UCRT64

Instalando el Compilador GCC a 64bits:

pacman -S mingw-w64-ucrt-x86_64-gcc

```
M ~
oscar@WIN-10 UCRT64 ~
$ pacman -S mingw-w64-ucrt-x86_64-gcc
resolving dependencies...
looking for conflicting packages...

Packages (16) mingw-w64-ucrt-x86_64-binutils-2.44-1
               mingw-w64-ucrt-x86_64-crt-git-12.0.0.r657.g1e8b1ccdd-1
               mingw-w64-ucrt-x86_64-gcc-libs-14.2.0-3  mingw-w64-ucrt-x86_64-gettext-runtime-0.24-1
               mingw-w64-ucrt-x86_64-gmp-6.3.0-2
               mingw-w64-ucrt-x86_64-headers-git-12.0.0.r657.g1e8b1ccdd-1
               mingw-w64-ucrt-x86_64-isl-0.27-1  mingw-w64-ucrt-x86_64-libiconv-1.18-1
               mingw-w64-ucrt-x86_64-libwinpthread-git-12.0.0.r657.g1e8b1ccdd-1
               mingw-w64-ucrt-x86_64-mpc-1.3.1-2  mingw-w64-ucrt-x86_64-mpfr-4.2.2-1
               mingw-w64-ucrt-x86_64-windows-default-manifest-6.4-4
               mingw-w64-ucrt-x86_64-winthreads-git-12.0.0.r657.g1e8b1ccdd-1
               mingw-w64-ucrt-x86_64-zlib-1.3.1-1  mingw-w64-ucrt-x86_64-zstd-1.5.7-1
               mingw-w64-ucrt-x86_64-gcc-14.2.0-3

Total Download Size:    65.85 MiB
Total Installed Size:  519.33 MiB

:: Proceed with installation? [Y/n] Y
:: Retrieving packages...
mingw-w64-ucrt-x86_64-isl-0.27-1 1452.5 KiB 105 KiB/s 00:04 [#####] 100%

(13/16) installing mingw-w64-ucrt-x86_64-mpc [#####] 100
(14/16) installing mingw-w64-ucrt-x86_64-windows-default-manifest [#####] 100
(15/16) installing mingw-w64-ucrt-x86_64-winthreads-git [#####] 100
(16/16) installing mingw-w64-ucrt-x86_64-gcc [#####] 100% 0

oscar@WIN-10 UCRT64 ~
$
```

Instalando paquetes con el Comando pacman en el Terminal UCRT64

Instalando el Toolchain completo (Cadena de Herramientas) para el Compilador GCC:

pacman -S mingw-w64-ucrt-x86_64-toolchain

```
M ~
oscar@WIN-10 UCRT64 ~
$ pacman -S mingw-w64-ucrt-x86_64-toolchain
:: There are 13 members in group mingw-w64-ucrt-x86_64-toolchain:
:: Repository ucrt64
 1) mingw-w64-ucrt-x86_64-binutils  2) mingw-w64-ucrt-x86_64-crt-git
 3) mingw-w64-ucrt-x86_64-gcc      4) mingw-w64-ucrt-x86_64-gdb
 5) mingw-w64-ucrt-x86_64-gdb-multiarch 6) mingw-w64-ucrt-x86_64-headers-git
 7) mingw-w64-ucrt-x86_64-libmangle-git 8) mingw-w64-ucrt-x86_64-libwinpthread-git
 9) mingw-w64-ucrt-x86_64-make      10) mingw-w64-ucrt-x86_64-pkgconf
11) mingw-w64-ucrt-x86_64-tools-git 12) mingw-w64-ucrt-x86_64-winpthread-git
13) mingw-w64-ucrt-x86_64-winstorecompat-git

Enter a selection (default=all):
warning: mingw-w64-ucrt-x86_64-binutils-2.44-1 is up to date -- reinstalling
warning: mingw-w64-ucrt-x86_64-crt-git-12.0.0.r657.g1e8b1ccdd-1 is up to date -- reinstalling
warning: mingw-w64-ucrt-x86_64-gcc-14.2.0-3 is up to date -- reinstalling
(27/30) installing mingw-w64-ucrt-x86_64-make [#####] 100%
(28/30) installing mingw-w64-ucrt-x86_64-pkgconf [#####] 100%
(29/30) installing mingw-w64-ucrt-x86_64-tools-git [#####] 100%
(30/30) installing mingw-w64-ucrt-x86_64-winstoreco... [#####] 100%

oscar@WIN-10 UCRT64 ~
$
```

Instalando paquetes con el Comando pacman en el Terminal UCRT64

Instalando NEOVIM

pacman -Ss neovim

pacman -S mingw-w64-ucrt-x86_64-neovim



Neovim. (s.f.)

Instalando NANO

pacman -Ss nano

pacman -S nano

pacman -S nano-syntax-highlighting

^ = Ctrl
M = Alt



GNU nano. (s.f.)

Instalando paquetes con el Comando pacman en el Terminal UCRT64



wxWidgets
Cross-Platform GUI Library

Instalando WXWIDGETS:

pacman -S mingw-w64-ucrt-x86_64-wxWidgets

```
oscar@WIN-10 UCRT64 ~
```

```
$ pacman -S mingw-w64-ucrt-x86_64-wxwidgets
```

```
resolving dependencies...
```

```
looking for conflicting packages...
```

```
warning: dependency cycle detected:
```

```
warning: mingw-w64-ucrt-x86_64-libwebp will be installed before its mingw-w64-ucrt-x86_64-libtiff dependency
```

```
Packages (10) mingw-w64-ucrt-x86_64-giflib-5.2.2-1  mingw-w64-ucrt-x86_64-jbigkit-2.1-5  
mingw-w64-ucrt-x86_64-lerc-4.0.0-1  mingw-w64-ucrt-x86_64-libdeflate-1.23-1  
mingw-w64-ucrt-x86_64-libjpeg-turbo-3.1.0-1  
mingw-w64-ucrt-x86_64-libpng-1.6.47-1  mingw-w64-ucrt-x86_64-libtiff-4.7.0-1  
mingw-w64-ucrt-x86_64-libwebp-1.5.0-1  
mingw-w64-ucrt-x86_64-wxwidgets3.0-msw-libs-3.0.5.1-16  
mingw-w64-ucrt-x86_64-wxwidgets3.0-msw-3.0.5.1-16
```

```
Total Download Size: 19.09 MiB
```

```
Total Installed Size: 143.09 MiB
```

```
:: Proceed with installation? [Y/n] |
```


Instalando paquetes con el Comando pacman en el Terminal UCRT64

Instalando WXWIDGETS:

pacman -Ss wxwidgets



wxWidgets
Cross-Platform GUI Library

pacman -S mingw-w64-ucrt-x86_64-wxPython

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-common

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-common-libs

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-gtk3

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-gtk3-libs

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-msw

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-msw-cb_headers

pacman -S mingw-w64-ucrt-x86_64-wxwidgets3.2-msw-libs

Instalando paquetes con el Comando pacman en el Terminal UCRT64

Instalando PAQUETES COMPRESORES

pacman -Ss 7z

pacman -S mingw-w64-ucrt-x86_64-7zip

pacman -S mingw-w64-ucrt-x86_64-ofarc

pacman -S zip

pacman -S unzip



7-Zip. (s.f.)

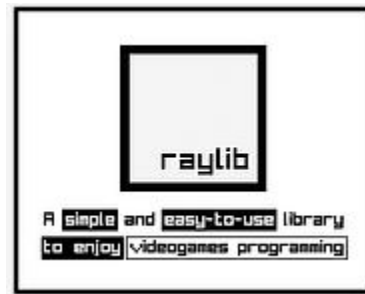
Instalando paquetes con el Comando pacman en el Terminal UCRT64

Biblioteca de programación de juegos sencilla y fácil de usar
(mingw-w64)

Instalando RAYLIB :

pacman -Ss raylib

pacman -S mingw-w64-ucrt-x86_64-raylib



Biblioteca portátil de código abierto para importar
de forma uniforme diversos formatos de modelos
3D conocidos (mingw-w64)

Instalando ASSIMP :

pacman -Ss assimp

pacman -S mingw-w64-x86_64-assimp



Raylib (2024)

Comando pacman (Gestor de Paquetes)

Si se presenta errores al instalar toolchain como:

Errors occurred, no packages were upgraded. Solución:

pacman -S mingw-w64-ucrt-x86_64-toolchain --overwrite '*'

Fallo en la clave GPG:

error: mingw-w64-ucrt-x86_64-gcc: signature from ... is invalid

pacman-key --init

pacman-key --refresh-keys

Error 404 (archivo no encontrado):

error: failed retrieving file 'xxx.pkg.tar.zst' from mirror

pacman -Syyu (Fuerza la actualización completa de los mirrors)

Para instalar paquete:

pacman -S paquete

Para borrar paquete:

pacman -R paquete

Para borrar paquete con sus dependencias:

pacman -Rns paquete

Para limpiar cache:

pacman -Scc

pacman -Syyu

Para ver todo los paquetes:

pacman -Si

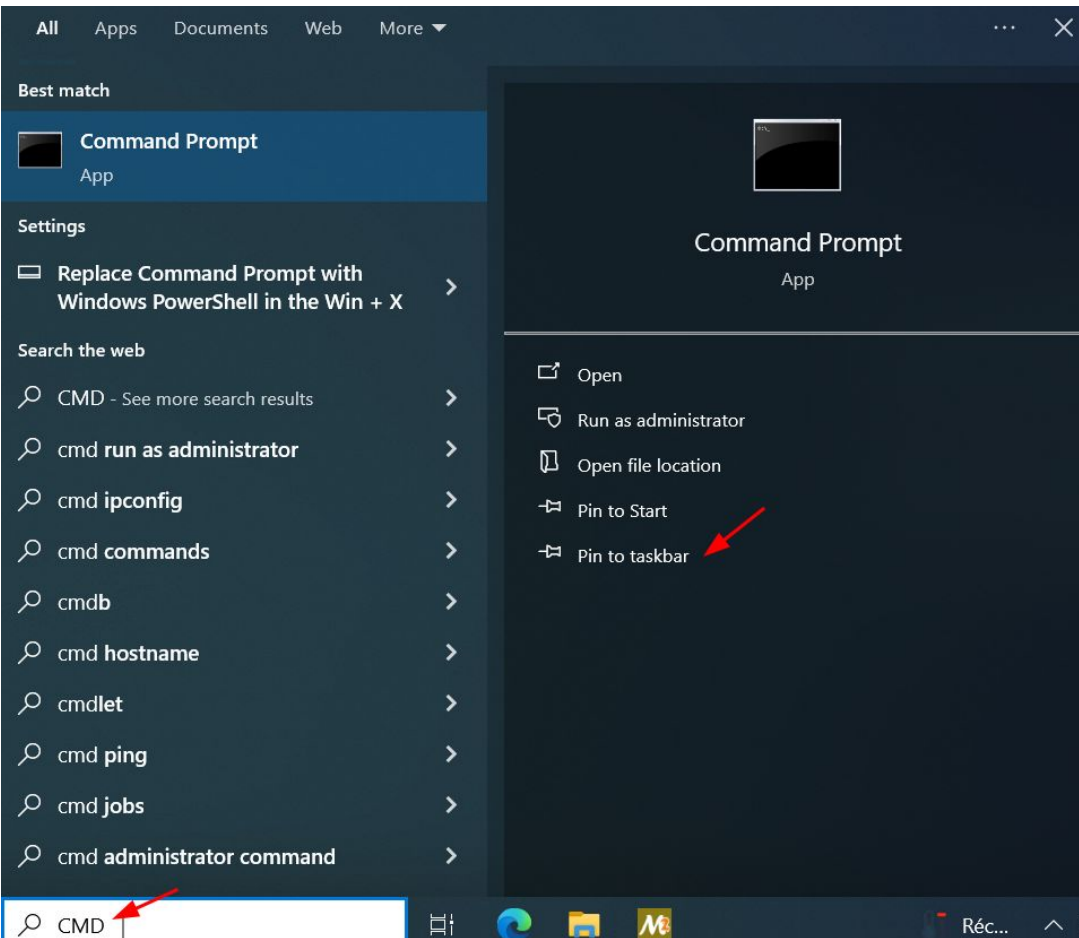
Para buscar paquete:

pacman -Ss paquete

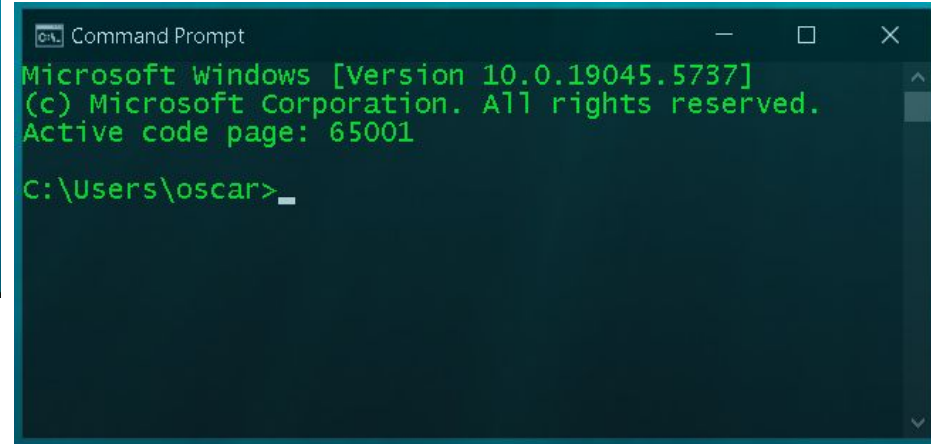
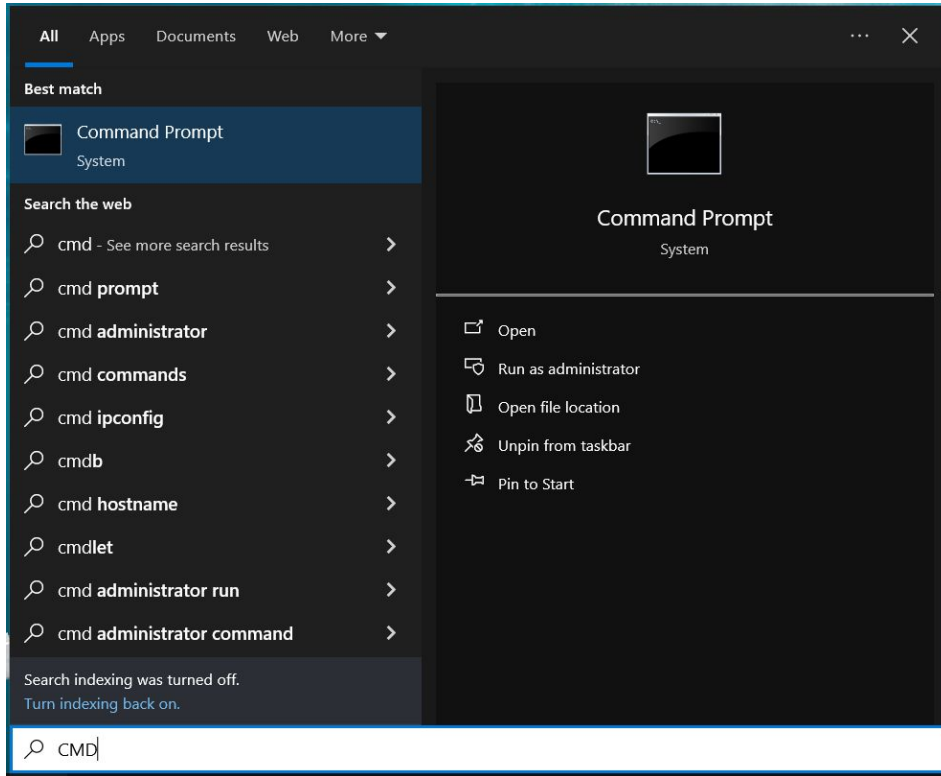
Para ver paquetes instalados:

pacman -Q

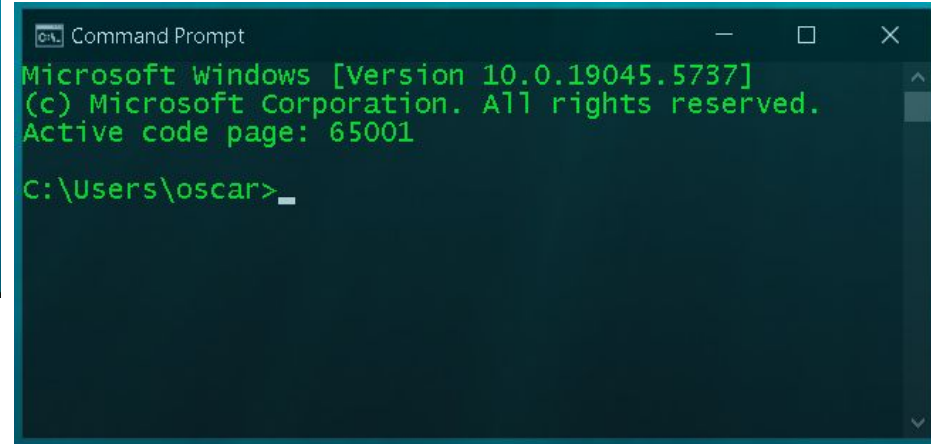
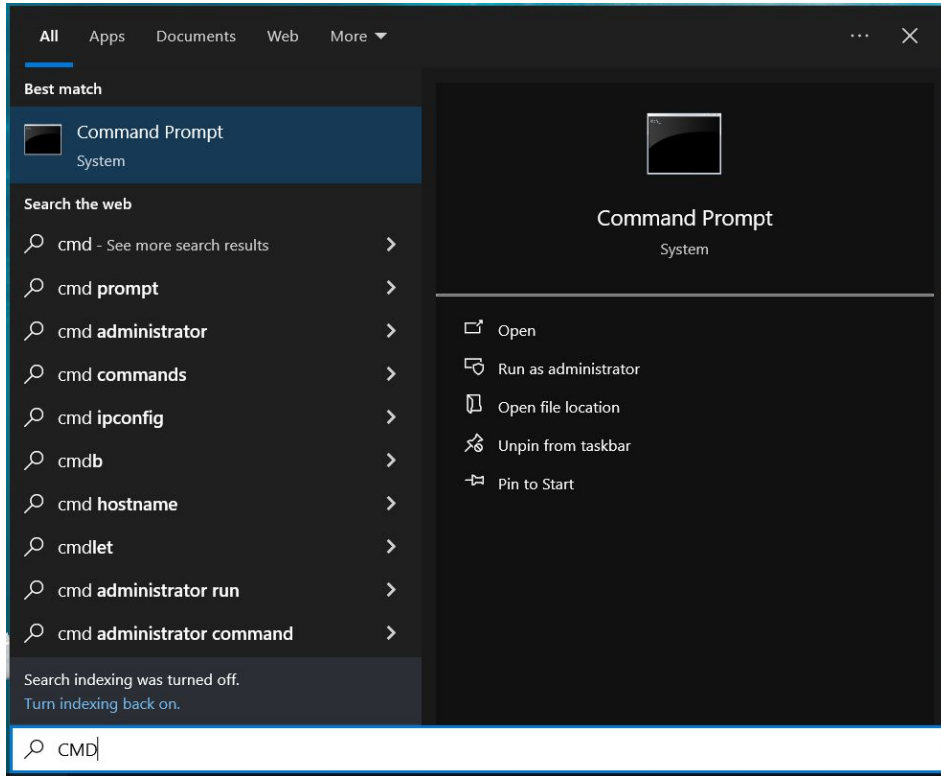
Terminal CMD



Terminal CMD



Terminal CMD



%SystemRoot%\system32

%SystemRoot%

%SystemRoot%\System32\Wbem

%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\

%SYSTEMROOT%\System32\OpenSSH\

C:\prg\msys64\ucrt64\bin

C:\Program Files\Notepad++

C:\Program Files\7-Zip

C:\prg\bin

New

Edit

Browse...

Delete

Move Up

Move Down

Edit text...

OK

Cancel

El PATH de Windows

En CMD:

SystemPropertiesAdvanced

C:\prg\msys64\ucrt64\bin

C:\Program Files\Notepad++

C:\Program Files\7-Zip

C:\prg\bin

En CMD:

El PATH de Windows

SystemPropertiesAdvanced → Environment Variables → Path → Edit → Edit Text ...

El Path es de: System Variables

Edit System Variable ×

Variable name: Path

Variable value: %SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEM

Browse Directory... Browse File... OK Cancel

%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0;%SYSTEMROOT%\System32\OpenSSH\;

C:\prg\msys64\ucrt64\bin;C:\Program Files\Notepad++;C:\Program Files\7-Zip;C:\prg\bin;

El PATH de Windows

En el Terminal CMD digitamos: **SYSTEMPROPERTIESADVANCED**

Command Prompt

(c) Microsoft Corporation. All rights reserved.
Active code page: 65001
C:\Users\oscar>SYSTEMPROPERTIESADVANCED
C:\Users\oscar>

Environment Variables

User variables for oscar

Variable	Value
OneDrive	C:\Users\oscar\OneDrive
Path	C:\Users\oscar\AppData\Local\Microsoft\WindowsApps;
TEMP	C:\Users\oscar\AppData\Local\Temp
TMP	C:\Users\oscar\AppData\Local\Temp

New... Edit... Delete

System variables

Variable	Value
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	2
OS	Windows_NT
Path	C:\Windows\system32;C:\Windows;C:\Windows\System32\Wb...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTU...	AMD64
PROCESSOR_IDENTIFIER	00000000-0000-0000-0000-000000000000

New... Edit... Delete

System Properties

Computer Name Hardware Advanced System Protection Remote

You must be logged on as an Administrator to make most of these changes.

Performance

Visual effects, processor scheduling, memory usage, and virtual memory

Settings...

User Profiles

Desktop settings related to your sign-in

Settings...

Startup and Recovery

System startup, system failure, and debugging information

Settings...

Environment Variables...

OK Cancel Apply

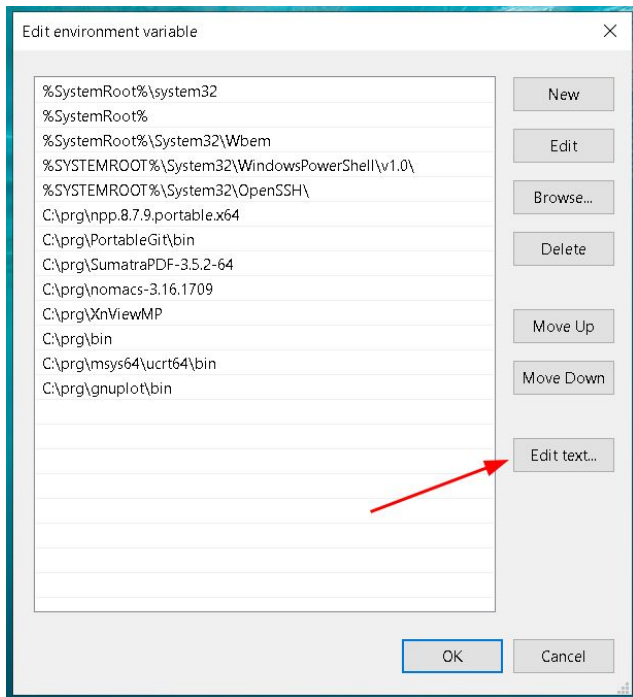
Edit environment variable

%SystemRoot%\system32
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\prg\hpp.8.7.9.portable.x64
C:\prg\PortableGit\bin
C:\prg\SumatraPDF-3.5.2-64
C:\prg\nomacs-3.16.1709
C:\prg\XnViewMP
C:\prg\bin
C:\prg\msys64\ucrt64\bin
C:\prg\gnuplot\bin

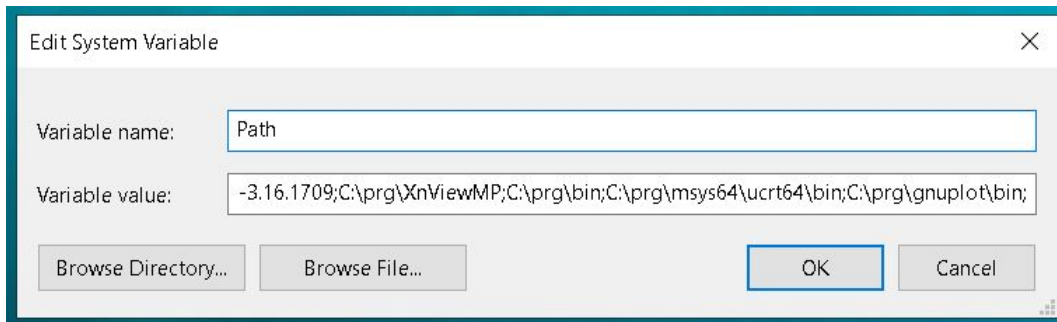
New Edit Browse... Delete Move Up Move Down Edit text...

OK Cancel

El PATH de Windows



C:\prg\msys64\ucrt64\bin



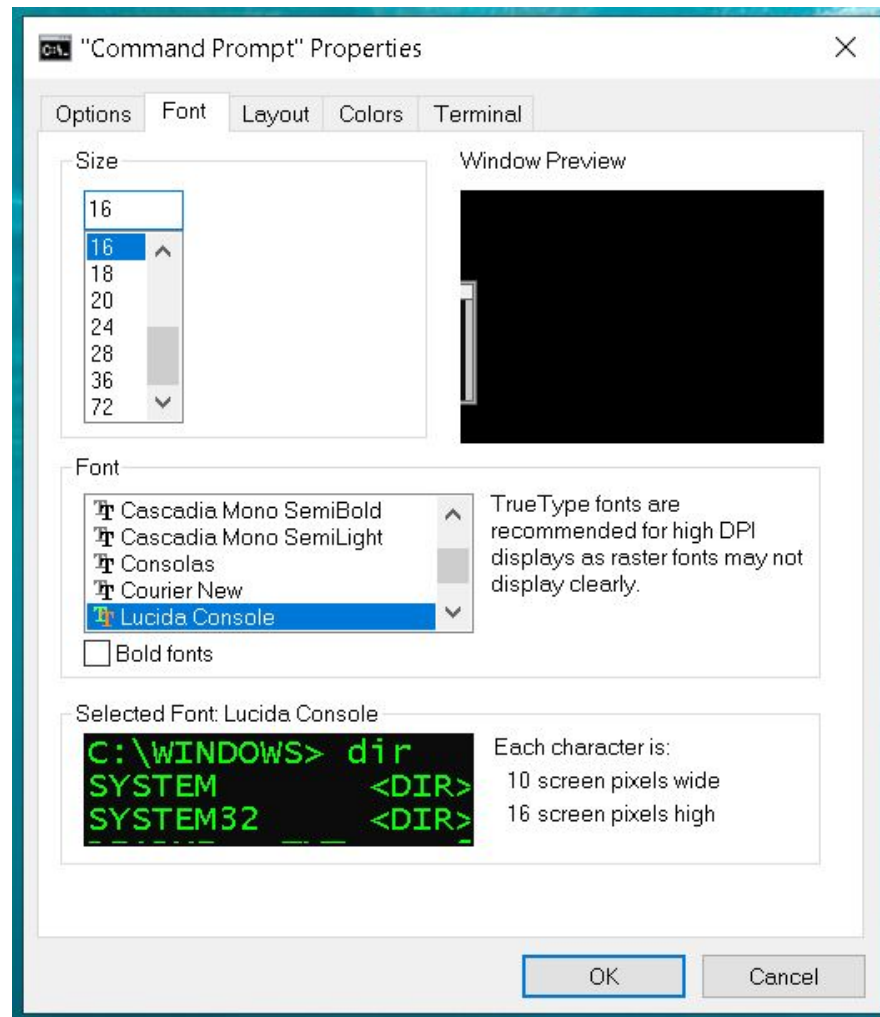
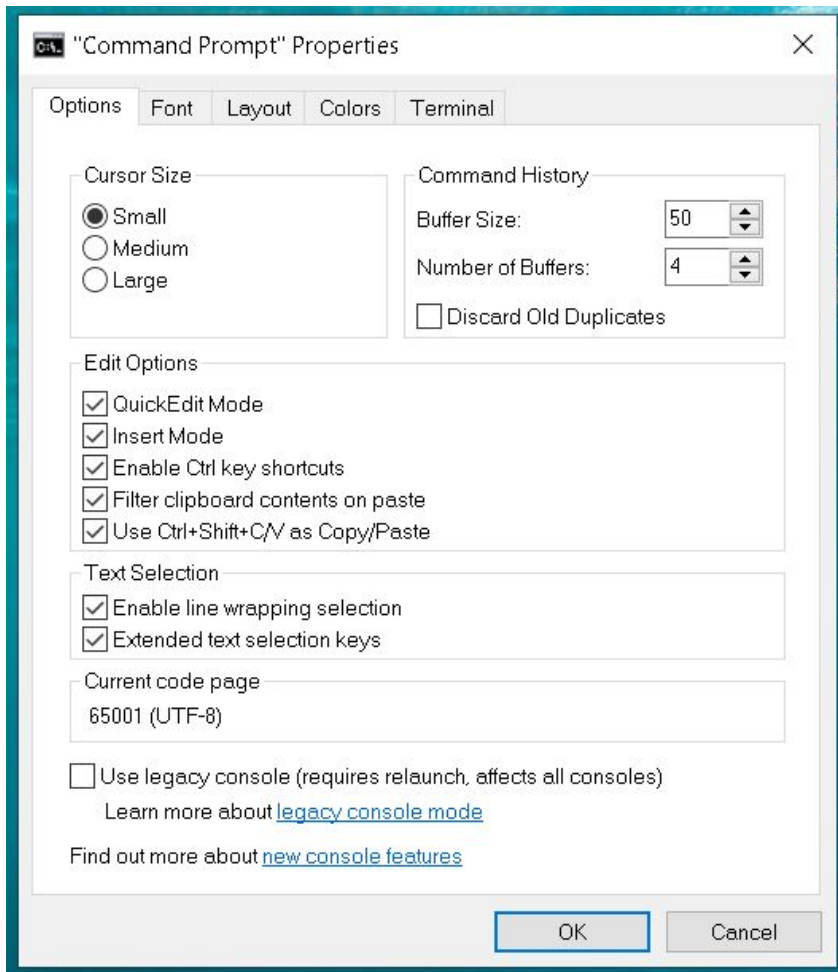
Las rutas de los diferentes campos están separadas por punto y coma (;). Siempre y terminan en punto y coma.

Esto permite pegar y copiar en bloque varias rutas a la vez. Hay que tener cuidado de no borrar las rutas que ya estaban en el sistema.

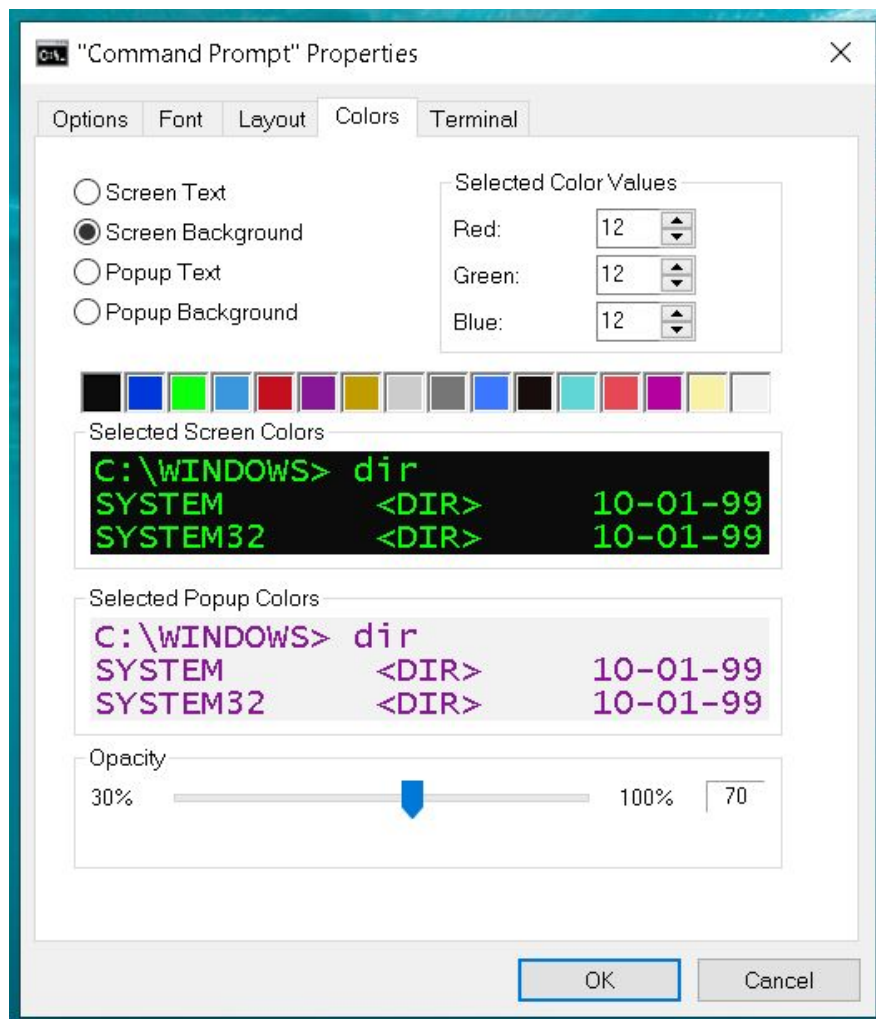
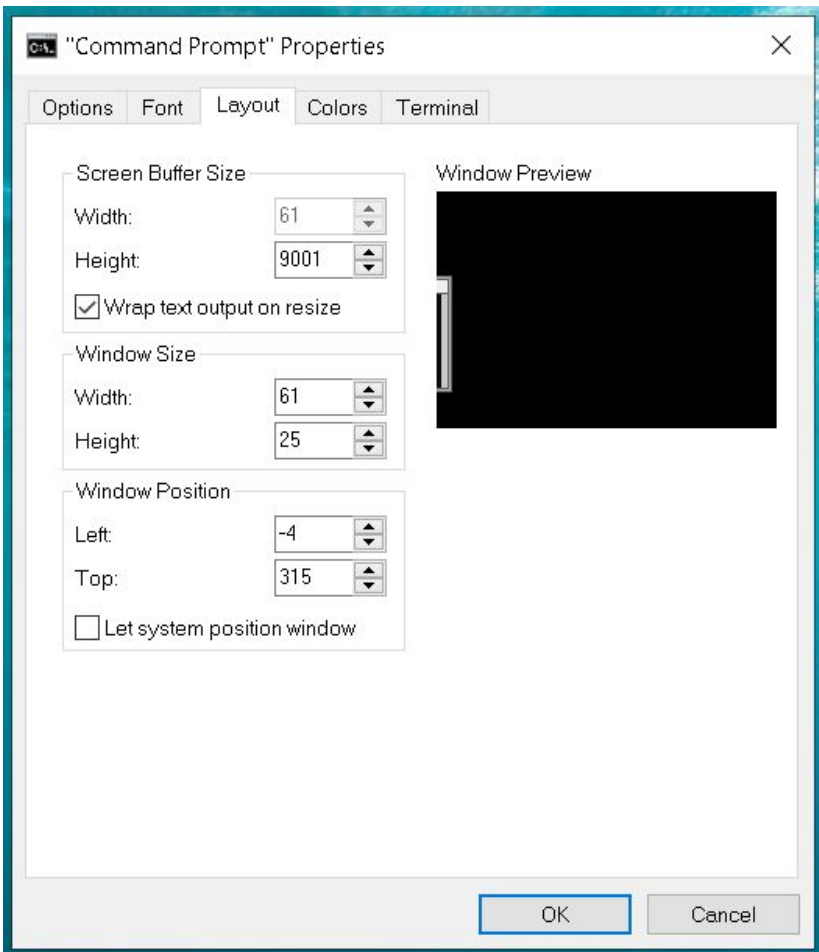
%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0%;%SYSTEMROOT%\System32\OpenSSH\;

C:\prg\npp.8.7.9.portable.x64;C:\prg\PortableGit\bin;C:\prg\SumatraPDF-3.5.2-64;C:\prg\nomacs-3.16.1709;C:\prg\XnViewMP;C:\prg\bin;C:\prg\msys64\ucrt64\bin;C:\prg\gnuplot\bin;

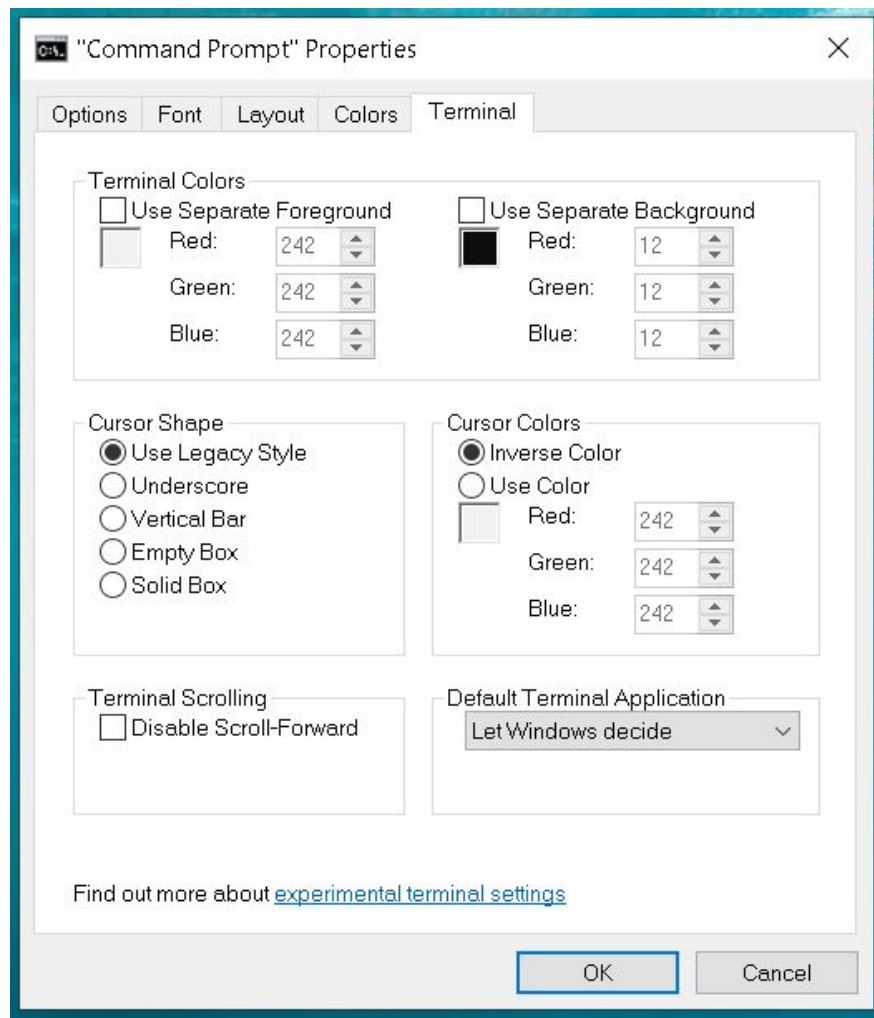
Terminal CMD



Terminal CMD



Terminal CMD



comando /? → para pedir ayuda.

dir → para ver contenido del directorio

control → para acceder al Panel de Control.

systemPropertiesAdvanced → Propiedades del Sistema. Para llegar al PATH.

copy archivo_origen archivo_destino → para copiar archivos.

mkdir nombre → para crear un directorio.

cls → para limpiar pantalla

cd directorio → para entrar dentro de un directorio.

cd c: o **cd ** → para ir a la partición C: de Windows

cd .. → para salir de un directorio.

cd ..\.. → para salir de dos directorios, uno dentro de otro.

del archivo → para borrar un archivo.

rd /s → para borrar un directorio con su contenido.

shutdown /s → para apagar el sistema windows.

ping web → para ver si una web esta operativa. Ejm. **ping google.com**

tasklist → para mostrar servicios ejecutándose en ese preciso momento.

ipconfig /all → Para ver los Números IP de tu sistema.

- winver** → Muestra la Versión de Windows.
- osk** → Teclado Virtual.
- psr** → Grabadora de Pasos.
- cleanmgr** → Limpieza de Disco.
- computerdefaults** → Aplicaciones por defecto.
- timedate.cpl** → Fecha y hora.
- appwiz.cpl** → Desinstalación, cambio o reparación de programa.
- devmgmt.msc** → Gestor de dispositivos.
- diskmgmt.msc** → Gestor de disco duro.
- powercfg.cpl** → Personalizar la gestión de energía.
- dxdiag** → Herramienta de diagnóstico.
- msinfo32** → Sistema de Información
- msconfig** → Configuración de sistema.
- regedit** → Editor de Registro.
- chcp 65001** → Configura temporalmente la codificación en UTF-8.

ROBOCOPY /E ORIGEN DESTINO

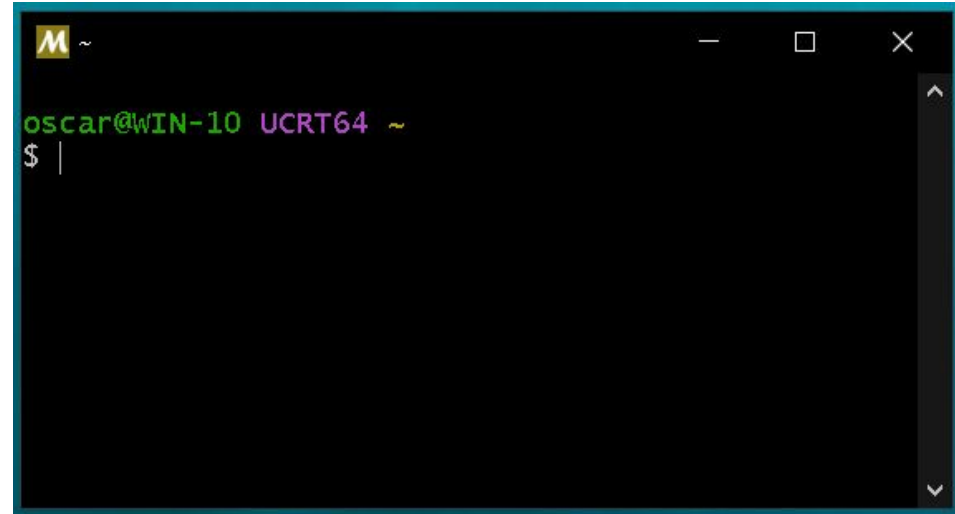
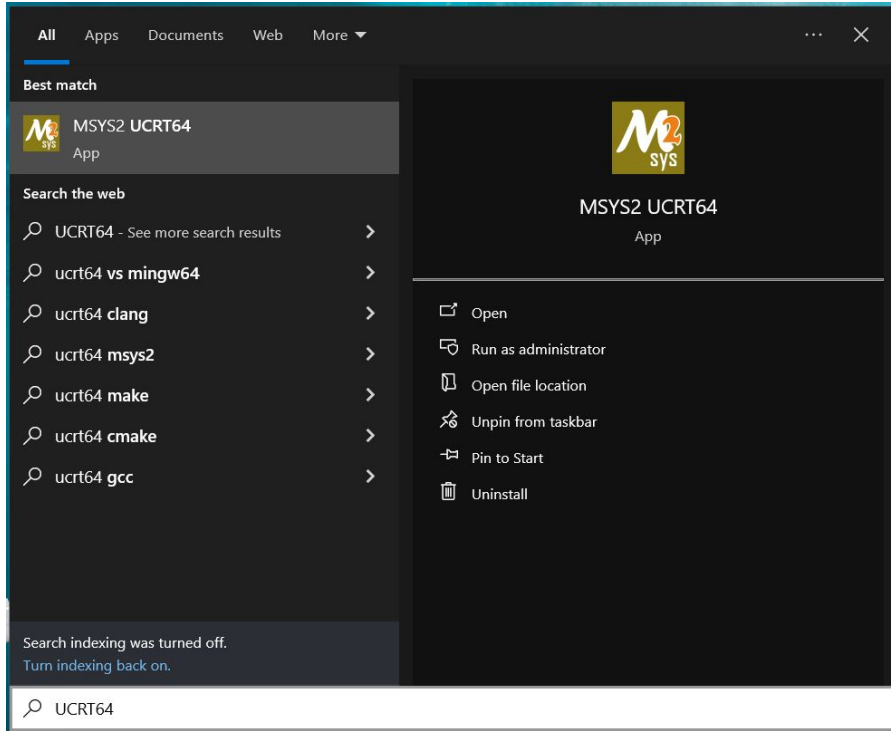
Comando DOS



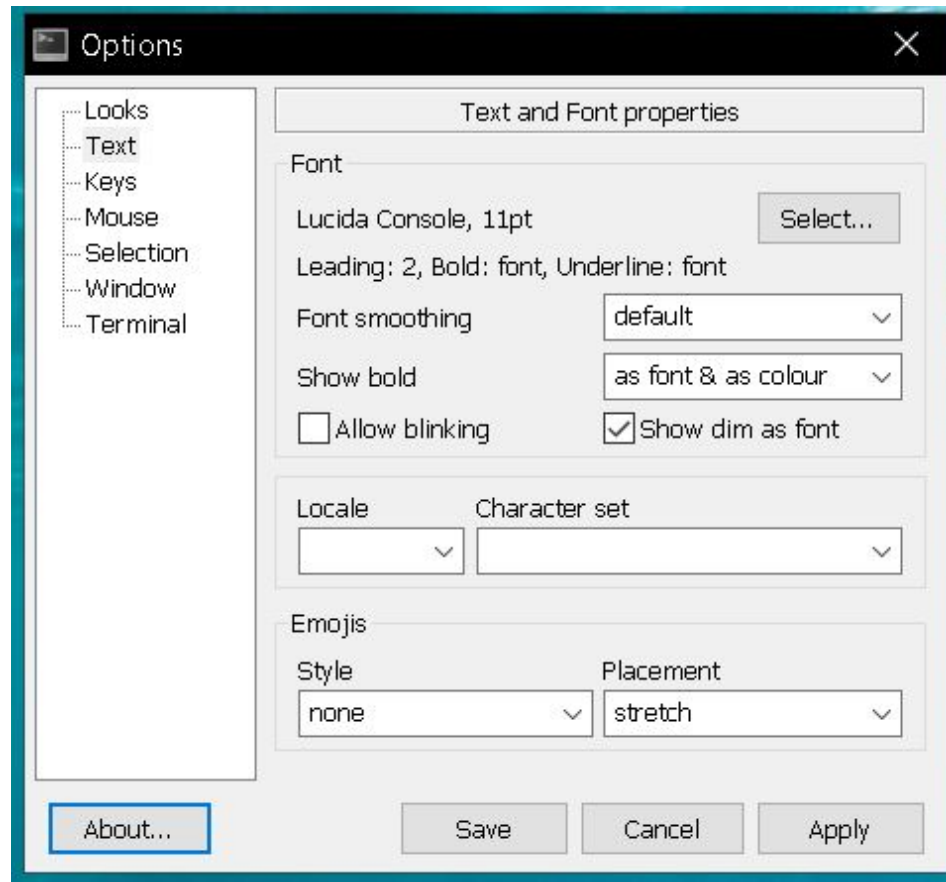
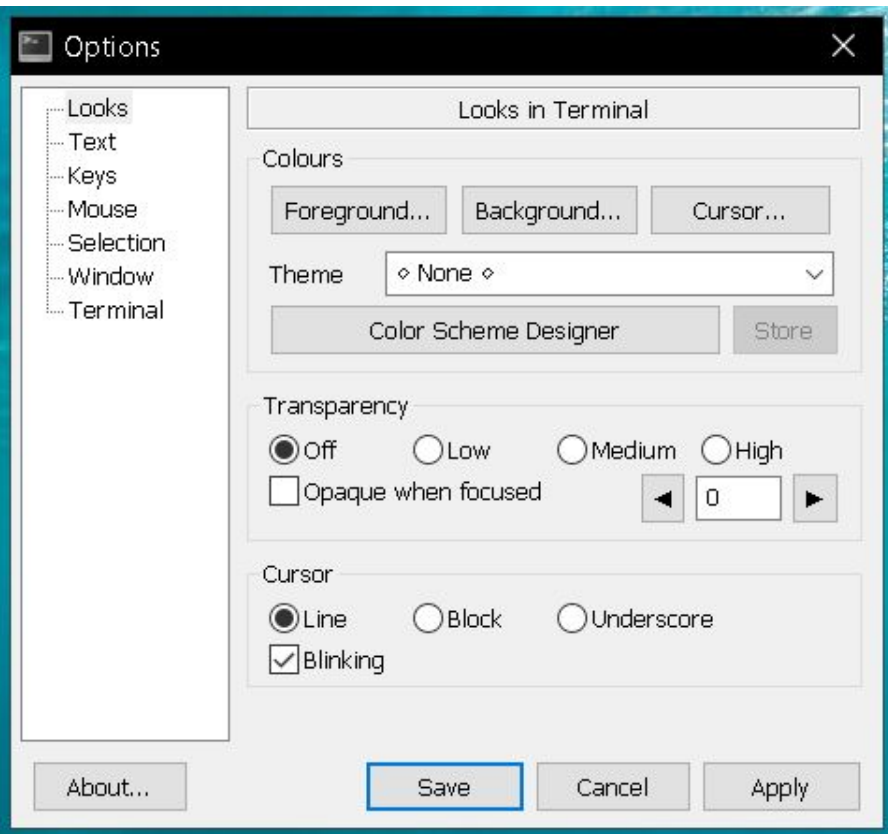
robocopy /?



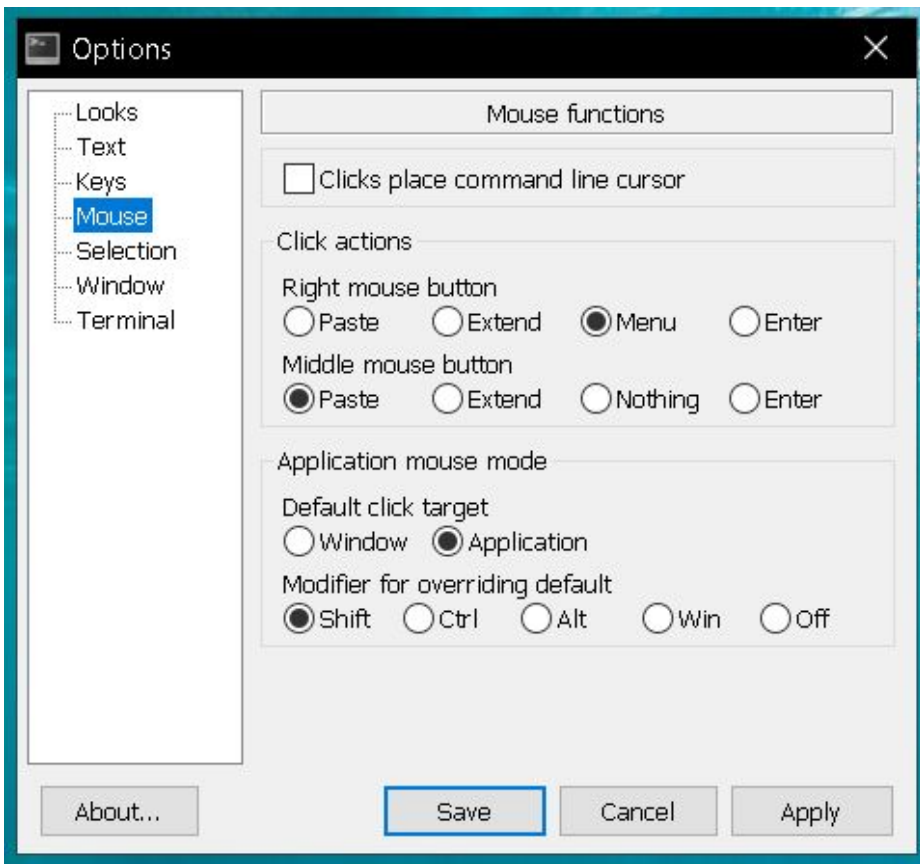
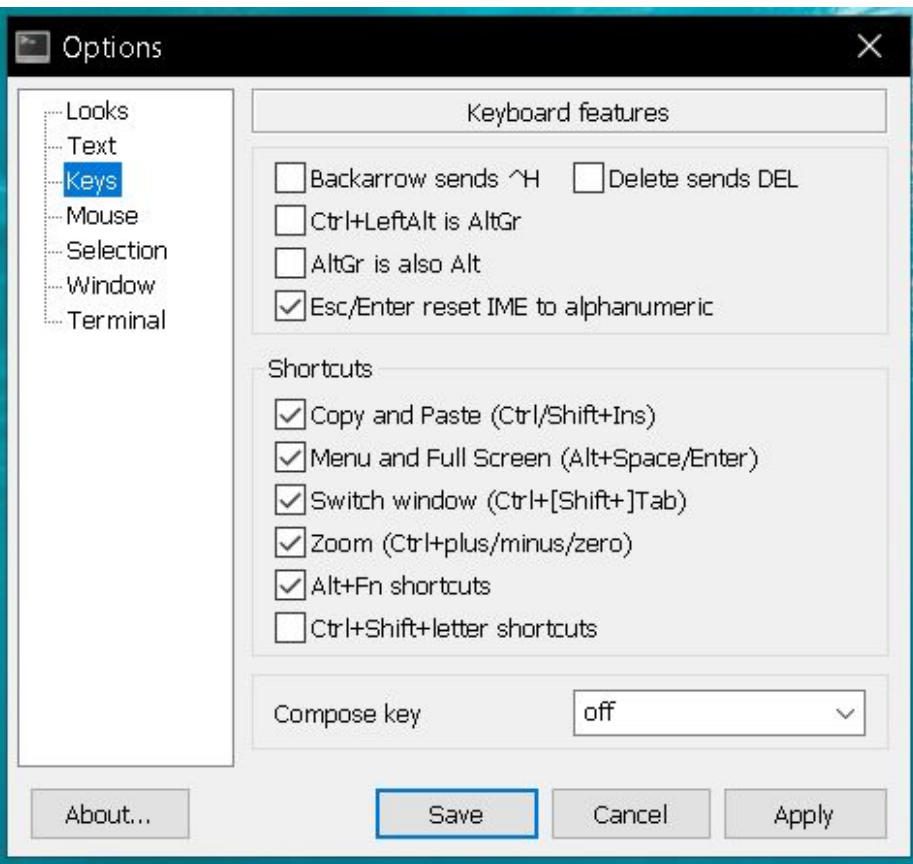
Terminal UCRT64



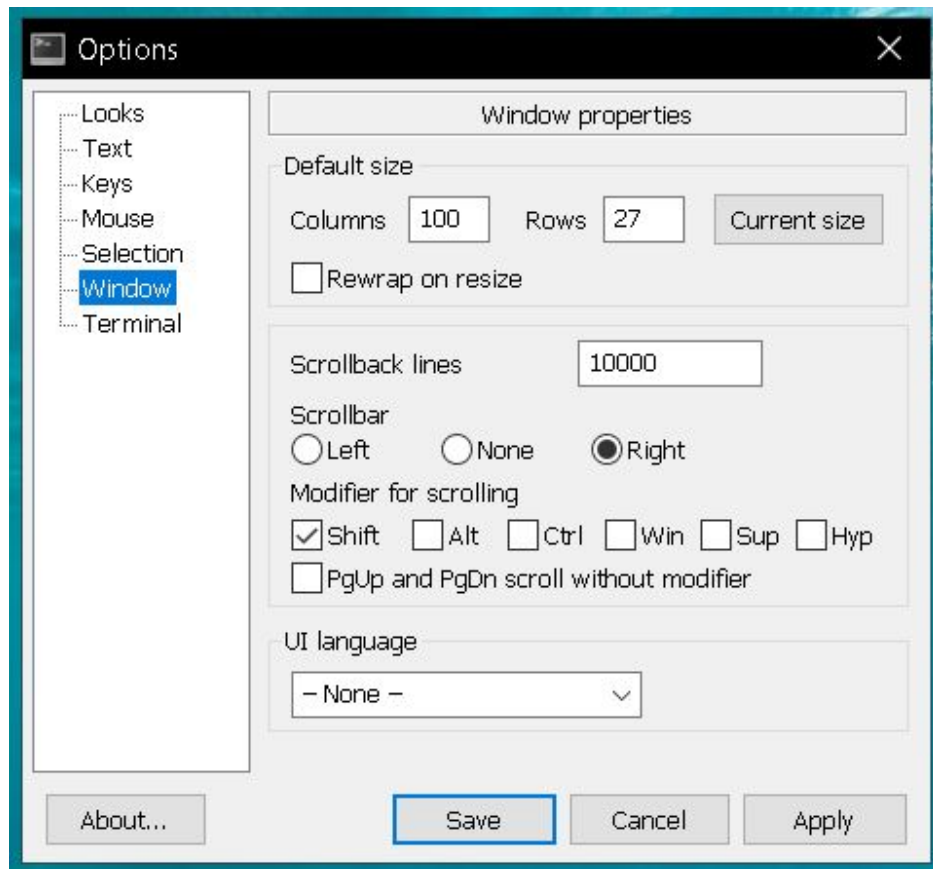
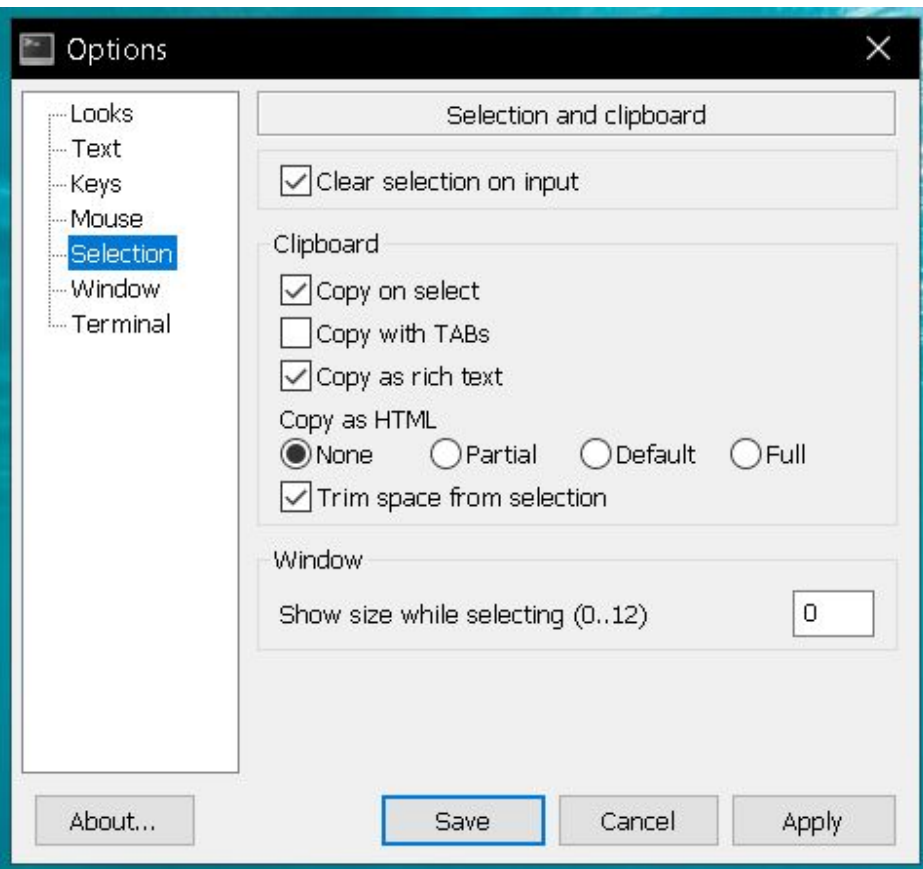
Terminal UCRT64



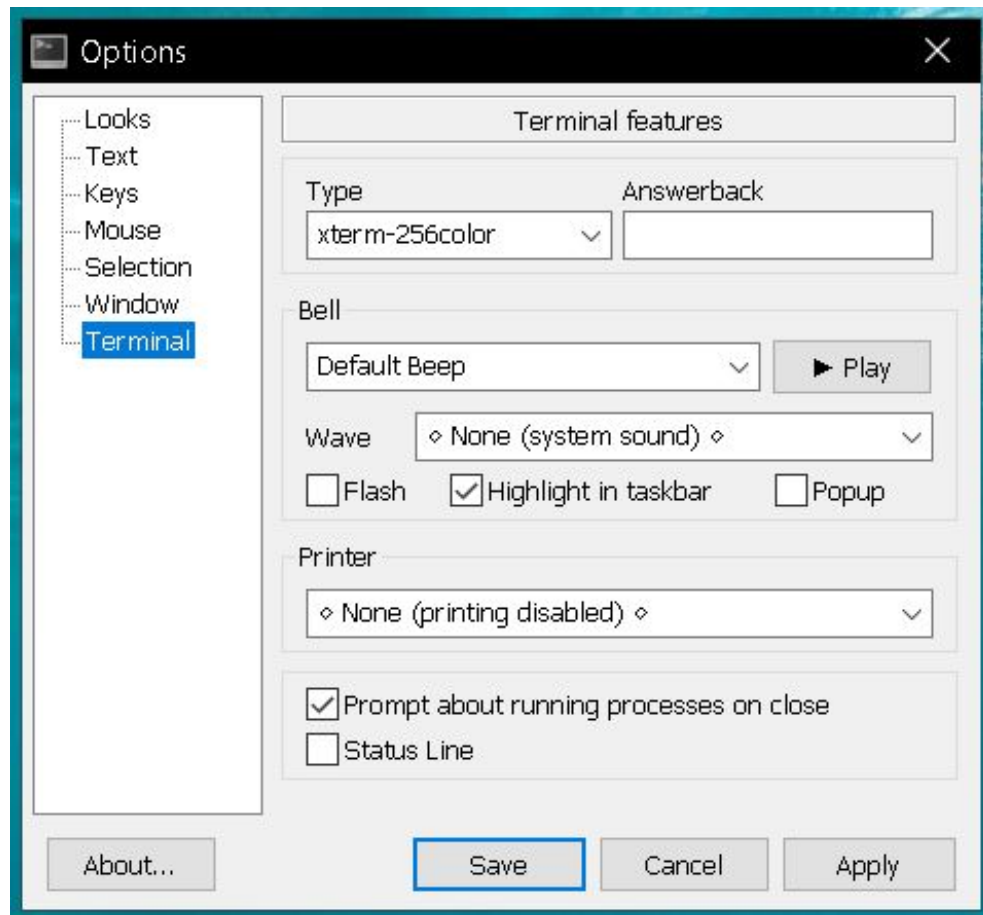
Terminal UCRT64



Terminal UCRT64



Terminal UCRT64



comando --help → para pedir ayuda.

ls -all → para ver contenido del directorio, inclusive archivos ocultos.

cat archivo → para el interior de un archivo de texto.

pwd → para ver la ruta actual.

cp origen destino → para copiar archivos y directorios.

mkdir nombre → para crear un directorio.

clear → para limpiar pantalla

cd directorio → para entrar dentro de un directorio.

cd /c/ → para entrar en la **partición C:** de Windows o cualquier otra. **cd /c**

cd .. → para salir de un directorio.

cd ../.. → para salir de dos directorios, uno dentro de otro.

rm -rf directorio → Elimina archivo y/o directorios.

mv → para mover o renombrar archivos o directorios.

ping web → para ver si una web está operativa. Ejm. **ping google.com**

wc → para contar palabras y el número de bytes de un archivo.

nvim → para activar el editor de texto plano neovim.

nano → para activar el editor de texto plano nano.

df -h → para ver el uso del espacio en disco.

cal → para un calendario en el mes actual.

Programación por Lotes o Batch Programming en Windows (CMD)

@echo off

:: Mostrar mensaje

echo Hola, alumnos de Ingeniería.

echo Este es un ejemplo de un script por lotes en Windows.

:: Esperar entrada del usuario

pause

:: Mostrar la fecha y hora actuales

echo Fecha y hora actual:

date /T

time /T

:: Fin del script

echo Fin del programa.

pause

hola.bat

@echo off → Oculta los comandos del script.

echo → Muestra mensajes en pantalla.

pause → Detiene la ejecución hasta que el usuario presione una tecla.

date /T → Muestran la fecha

time /T → hora actual.

Programación por Lotes o Bash Programming en Linux (UCRT64)

```
#!/bin/bash
```

```
# Mostrar mensaje
```

```
echo "Hola, alumnos de  
Ingeniería."
```

```
echo "Este es un ejemplo de un  
script por lotes en Linux."
```

```
# Esperar entrada del usuario
```

```
read -p "Presiona ENTER para  
continuar..."
```

```
# Mostrar la fecha y hora actuales
```

```
echo "Fecha y hora actual:"  
date
```

```
# Fin del script
```

```
echo "Fin del programa."
```

hola.sh

`#!/bin/bash` → Indica que se usará el intérprete Bash.

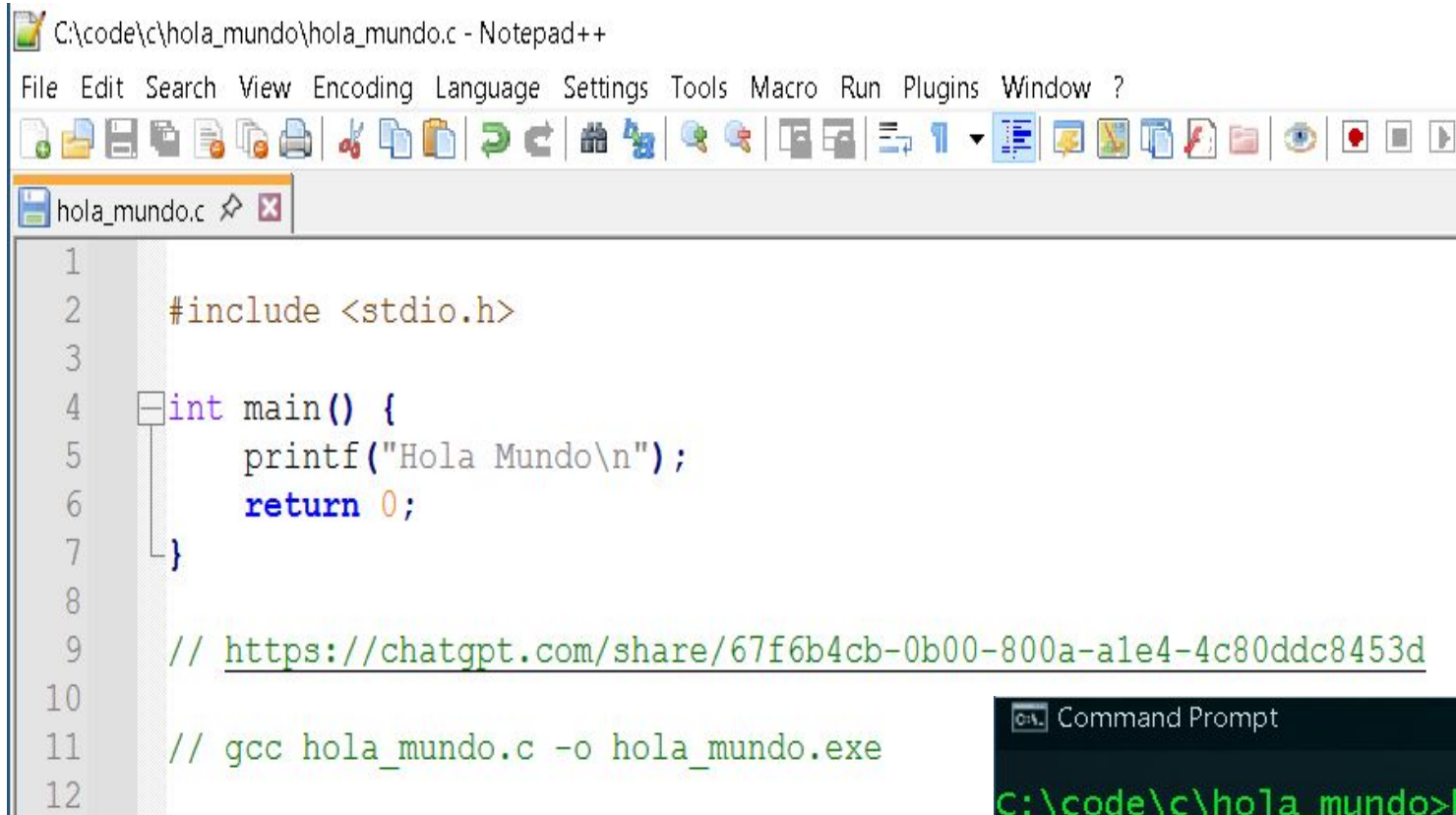
`echo` → Muestra texto en pantalla.

`read` → Espera una tecla o entrada.

`date` → Muestra la fecha y hora del sistema.

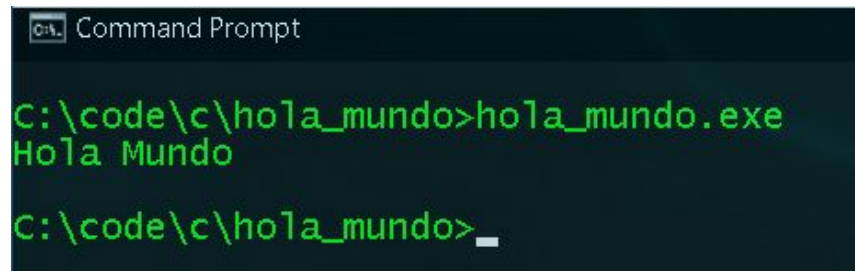
chmod +x hola.sh

Ejemplo 01: hola_mundo.c



The image shows a Notepad++ window titled "C:\code\c\hola_mundo\hola_mundo.c - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and ?. The toolbar contains various icons for file operations, editing, and development. The tab bar shows "hola_mundo.c". The code editor contains the following C program:

```
1
2  #include <stdio.h>
3
4  int main() {
5      printf("Hola Mundo\n");
6      return 0;
7  }
8
9  // https://chatgpt.com/share/67f6b4cb-0b00-800a-a1e4-4c80ddc8453d
10
11  // gcc hola_mundo.c -o hola_mundo.exe
12
```



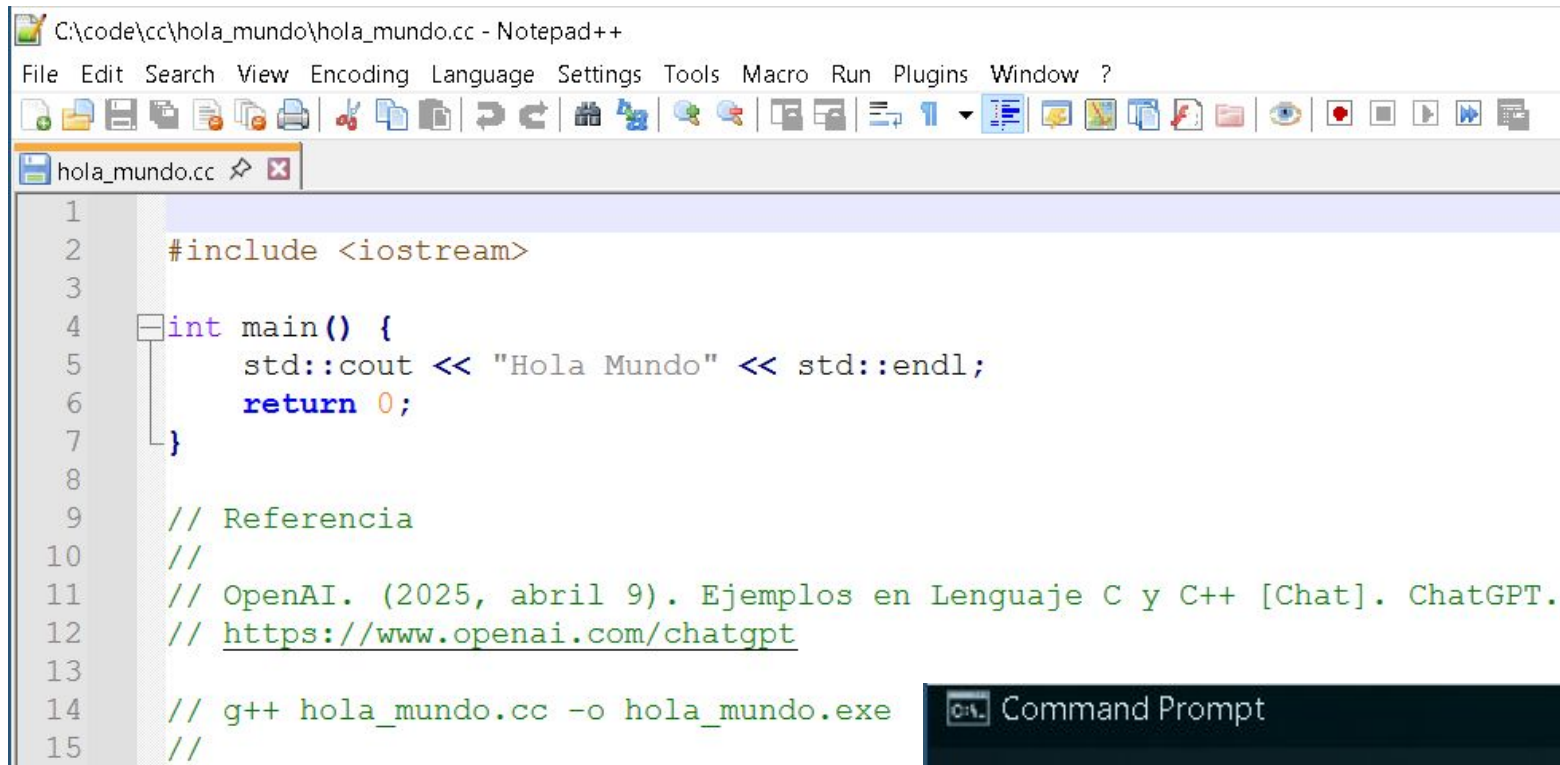
The image shows a Command Prompt window titled "C:\> Command Prompt". It displays the execution of the hola_mundo.exe program, which outputs "Hola Mundo".

```
C:\code\c\hola_mundo>hola_mundo.exe
Hola Mundo

C:\code\c\hola_mundo>
```

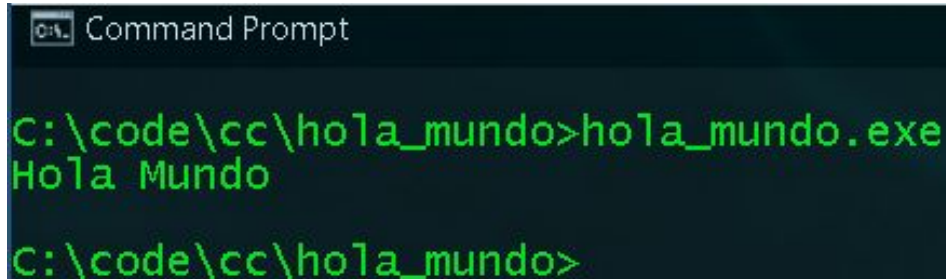

Ejemplo 01: hola_mundo.cc

Se puede trabajar con extensión: cc o cpp



The image shows a Notepad++ window titled "C:\code\cc\hola_mundo\hola_mundo.cc - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and ?. The toolbar contains various icons for file operations and editing. The code editor shows the following content:

```
1
2  #include <iostream>
3
4  int main() {
5      std::cout << "Hola Mundo" << std::endl;
6      return 0;
7  }
8
9  // Referencia
10 //
11 // OpenAI. (2025, abril 9). Ejemplos en Lenguaje C y C++ [Chat]. ChatGPT.
12 // https://www.openai.com/chatgpt
13
14 // g++ hola_mundo.cc -o hola_mundo.exe
15 //
```



The image shows a Command Prompt window titled "0% Command Prompt". The command prompt shows the execution of the program:

```
C:\code\cc\hola_mundo>hola_mundo.exe
Hola Mundo

C:\code\cc\hola_mundo>
```



Меня зовут Дана.

СПАСИБО

BIBLIOGRAFÍA

Perplexity AI. (2025, abril 4). *Importancia del lenguaje C y C++ en el mantenimiento industrial y sus aplicaciones estadísticas*. Respuesta generada por un asistente virtual.

<https://www.perplexity.ai/search/cual-es-la-importancia-del-len-KDFdfp.uTXWcFVp2F2moVw>

Raylib (2024, 26 Abr.). *Raylib projects showcase 2024* [Video]. YouTube.

<https://www.youtube.com/watch?v=GdIGE-eDlZg>

Reclut (s.f.). *¿Cuál es la historia de C++?* <https://reclut.com/cual-es-la-historia-de-c/>

Digital News Asia (2012, 6 Sep.). *Honoring the Life of Dennis Richie. UNIX co-creator and father of C.* [Foto]

<https://www.digitalnewsasia.com/digital-economy/honoring-the-life-of-dennis-ritchie-unix-co-creator-and-father-of-c>

Stroustrup (s.f.), *Welcome to Bjarne Stroustrup's homepage!*. <https://www.stroustrup.com/>

Programador X (2024, 16 Abr.). *Aprende C en 45 minutos*. [Video]. YouTube

<https://www.youtube.com/watch?v=1-qjMS6C960>

Programación ATS (2016. 21 Feb.). *Programación en C - Presentación del curso*. [Video]. YouTube.

https://www.youtube.com/watch?v=5O1srQNyJXo&list=PLWtYZ2ejMVJmUTNE2QVaCd1y_6GslOeZ6

Alejandro Taboada Sanchez (2025). *Programación*. <https://programacionats.newzenler.com/>

Picando Código (2023, 10 Oct.). *¿Qué es el lenguaje de programación C?* [Video]. YouTube.

https://www.youtube.com/watch?v=sy1AuLyHc5Q&list=PLO8lO9oepSLvAvnxCn_uCsFVCAAYCDsGTK

BIBLIOGRAFÍA

Estudia con Marisol (2018, 21 May.) *¿Qué es CODIFICAR, CÓDIGO FUENTE y LENGUAJE MÁQUINA? - Lenguajes de Programación #1* [Video]. YouTube.

https://www.youtube.com/watch?v=fkiOskQbewM&list=PL46-B5QR6sHmlim_g4RzKqnH_5AIG7wP1

Lucas Moy (2022, 30 Nov.). *Curso de Programación en C (Desde 0 actualizado a 2023)* [Video]. ATL Academy. YouTube.

https://www.youtube.com/watch?v=0OTpkPo__TY

Código Facilito (2017, 9 Ene.). *Curso Básico de C - Introducción* [Video]. YouTube.

https://www.youtube.com/watch?v=rEsSxd0L4GI&list=PLpOqH6AE0tNgqknxjMAJ8bX_L1a7lnBaH

Programación ATS (2018, 2 Abr.). *Aprende Programación en C++* [Video]. YouTube.

https://www.youtube.com/watch?v=dJzLmjSJc2c&list=PLWtYZ2ejMVJIUu1rEHLCOi_oibctkl0Vh

Aprende a Programar (2024, 2 Feb.). *Curso Maestro de C++ De Cero a Programador Master #1* [Video]. YouTube.

<https://www.youtube.com/watch?v=RGSEnAzS4XY&list=PL5YuOOJ-FGGjjK8j50TkEcjCcZ8dRAiPy>

Pildoras Informáticas (2022, 10 Sep.). *Curso C++. ¿Por qué aprender C++? Vídeo 1* [Video]. YouTube.

<https://www.youtube.com/watch?v=EaTeni7YCyM&list=PLU8oAlHdN5BmoxxnllWdZfRcffMPCNck5>

Lucas Moy (2023, 30 May.). *Curso Completo de C++ para Principiantes (2023)* [Video]. ATL Academy. YouTube.

<https://www.youtube.com/watch?v=VQo6gj7-hw8>

BIBLIOGRAFÍA

- Goin, M. y Lovos, E. (2020, 3 Set.). *Problemas y Algoritmos: Un enfoque práctico*. Colección: Lecturas de Cátedra. pp. 120. ISBN: 978-987-4960-31-3
https://editorial.unrn.edu.ar/index.php/catalogo/346/view_bl/62/lecturas-de-catedra/94/problemas-y-algoritmos-un-enfoque-practico?tab=getmybooksTab&is_show_data=1
- Goin, M. (2022, 11 May.). *Caminando junto al Lenguaje C*. Colección: Lecturas de Cátedra. pp. 188. ISBN: 978-987-4960-76-4
https://editorial.unrn.edu.ar/index.php/catalogo/346/view_bl/62/lecturas-de-catedra/26/caminando-junto-al-lenguaje-c?tab=getmybooksTab&is_show_data=1
- Ruiz-Linares, A. E. (2023, 28 Nov.). *Programación en C para Ingenieros electrónicos*. Tecnológico Nacional de México. Instituto Tecnológico de Minatitlán. <https://rinacional.tecnm.mx/handle/TecNM/6243>
- Del Prado, A. M. (s.f.). *Lenguaje de Programación en C. Apuntes de Catedra. Informática*. Universidad Nacional de Catamarca – Secretaría de Ciencia y Tecnología Editorial Científica Universitari. a ISBN: 978-987-661-127-5
<https://editorial.unca.edu.ar/Publicacione%20on%20line/CUADERNOS%20DE%20CATEDRA/Ana%20Maria%20del%20Prado/ApunteDeCatedraInformatica2.pdf>
- Guerra, J. E., Ramos, M. V. y Vallejo, G. E. (2023, 22). *Programando en C desde la práctica: Problemas Resueltos*. Puerto Madero Editorial. <https://puertomaderoeditorial.com.ar/index.php/pmea/catalog/book/36>
- Muñoz-Guerrero, L. E. (2021). *Introducción a la programación con C*. Editorial CIMTED. [Descargar PDF].
<https://memoriascimted.com/wp-content/uploads/2022/04/Libro-Introduccion-a-la-programacion-con-C.pdf>

BIBLIOGRAFÍA

- ASNetworking (2025, 12 May.).COPIA archivos y carpetas con MÁXIMO rendimiento con este comando.
https://editorial.unrn.edu.ar/index.php/catalogo/346/view_bl/62/lecturas-de-catedra/94/problemas-y-algoritmos-un-enfoque-practico?tab=getmybooksTab&is_show_data=1
- Miguel Hernando Gutiérrez (2020, Feb.). *C++ estándar. Apuntes de Informática Industrial y Comunicaciones*. Escuela Técnica Superior de Ingeniería y Diseño Industrial.
<https://blogs.upm.es/miguelhernando/wp-content/uploads/sites/734/2022/11/libroC2020conPortada.pdf>
- Rodríguez-Losada, D y Hernando-Gutierrez, M. (2021). *Diseño y programación Orientados a Objetos. Una Visión Aplicada*. 2da. Revisión.
https://blogs.upm.es/miguelhernando/wp-content/uploads/sites/734/2022/11/DisenoyProgramacionOrientadaObjetos_UnaVisionAplicada_2021.pdf
- Gil-Espert. L. y Sánchez-Romero, M. (1999). *El C++ por la práctica: introducción al lenguaje y su filosofía*. Edicions UPC.
<https://upcommons.upc.edu/handle/2099.3/36408>
- Martínez del Río, F. (2019). *Fundamentos Básicos de Programación en C++*. Universidad de Jaén.
<https://www4.ujaen.es/~fmartin/apuntesC++.pdf>
- Uvidia, M. I., Villacrez, E. P. y Delgado J. E. (2024). *Programación Básica en C++*. Editorial UNACH. Universidad Nacional de Chimborazo. <https://editorial.unach.edu.ec/index.php/Editorial/catalog/book/253>
- Benjumea, V., y Roldán, M. (2024). *Fundamentos de Programación con el Lenguaje de Programación C++*. Universidad de Málaga. Recuperado de http://www.lcc.uma.es/~vicente/docencia/cppdoc/programacion_cxx.pdf

BIBLIOGRAFÍA

7-Zip. (s.f.). 7-Zip. <https://www.7-zip.org/>

WinMD5. (s.f.). *WinMD5 Free*. <https://www.winmd5.com/>

Notepad++ Team. (s.f.). *Notepad++*. <https://notepad-plus-plus.org/>

Gnuplot Project. (s.f.). *Gnuplot*. <http://www.gnuplot.info/>

Nomacs Project. (s.f.). *Nomacs – Image Lounge*. <https://nomacs.org/>

XnView Software. (s.f.). *XnView Software*. <https://www.xnview.com/en/>

Krzysztof Kowalczyk. (s.f.). *SumatraPDF*. <https://www.sumatrapdfreader.org/free-pdf-reader>

Git Project. (s.f.). *Git for Windows*. <https://git-scm.com/downloads/win>

GNU Project. (s.f.). *GCC, the GNU Compiler Collection*. <https://gcc.gnu.org/>

Clang Team. (s.f.). *Clang: a C language family frontend for LLVM*. <https://clang.org/>

Embarcadero Technologies. (s.f.). *Embarcadero*. <https://www.embarcadero.com/>

Microsoft. (s.f.). *Visual Studio: precios y suscripciones*. <https://visualstudio.microsoft.com/vs/pricing/?tab=paid-subscriptions>

BIBLIOGRAFÍA

IBM. (s.f.). *IBM Open XL C/C++ for AIX*. <https://www.ibm.com/products/open-xl-cpp-aix-compiler-power>

MinGW-w64 Project. (s.f.). *MinGW-w64*. <https://www.mingw-w64.org/>

MSYS2 Team. (s.f.). *MSYS2*. <https://www.msys2.org/>

wxWidgets Project. (s.f.). *wxWidgets: Cross-Platform GUI Library*. <https://www.wxwidgets.org/>

CMake Project. (s.f.). *CMake*. <https://cmake.org/>

Neovim Project. (s.f.). *Neovim*. <https://neovim.io/>

Davis E. King. (s.f.). *dlib C++ Library*. <https://dlib.net/>

GNU Project. (s.f.). *GNU Scientific Library (GSL)*. <https://www.gnu.org/software/gsl/>

Guennebaud, G., Jacob, B., y colaboradores. (s.f.). *Eigen*. <https://eigen.tuxfamily.org/>

mlpack Developers. (s.f.). *mlpack: fast, flexible machine learning library*. <https://www.mlpack.org/>

ImageMagick Studio LLC. (s.f.). *ImageMagick*. <https://imagemagick.org/index.php>

SDL Project. (s.f.). *Simple DirectMedia Layer*. <https://www.libsdl.org/>

BIBLIOGRAFÍA

raylib Technologies. (s.f.). *raylib: a simple and easy-to-use library to enjoy videogames programming*. <https://www.raylib.com/>

Wireshark Foundation. (s.f.). *Wireshark*. <https://www.wireshark.org/>

Perl Foundation. (s.f.). *The Perl Programming Language*. <https://www.perl.org/>

Curl Project. (s.f.). *cURL*. <https://curl.se/>

Python Software Foundation. (s.f.). *Welcome to Python.org*. <https://www.python.org/>

pygame Community. (s.f.). *pygame*. <https://www.pygame.org/>

Arma Developer Team. (s.f.). *ARMA*. <https://arma.sourceforge.net/>

OpenCV.org. (s.f.). *OpenCV*. <https://opencv.org/>

Leenissen, R. (s.f.). *Code::Blocks*. <https://leenissen.dk/>

Boost Community. (s.f.). *Boost C++ Libraries*. <https://www.boost.org/>

Microsoft. (s.f.). *Microsoft*. <https://www.microsoft.com/>

OpenAI (2025, 29 Abr.). *Script Bash para compilar y empaquetar aplicaciones wxWidgets con MSYS2-UCRT64 en Windows [Asistente conversacional, ChatGPT]*. <https://chat.openai.com/>

BIBLIOGRAFÍA

Torvalds, L. (s.f.). *Linux* [Repositorio en GitHub]. GitHub. <https://github.com/torvalds/linux>

Epic Games. (s.f.). *Unreal Engine*. Unreal Engine. <https://www.unrealengine.com/en-US>

GNU nano. (s.f.). *GNU nano editor*. <https://www.nano-editor.org/>

Neovim. (s.f.). *Neovim – hyperextensible Vim-based text editor*. <https://neovim.io/>

Geany. (s.f.). *Geany: The lightweight IDE*. <https://www.geany.org/>

GNUplot. (s.f.). *GNUplot: Command-driven interactive function plotting program*. <http://www.gnuplot.info/>

Sumatra PDF. (s.f.). *Sumatra PDF Reader: A free PDF reader for Windows*. <https://www.sumatrapdfreader.org/free-pdf-reader>

DW Cómo te afecta (2025, 23 Mayo). *Estos son los trabajos que la IA eliminará y hará mejor que tú* [Video]. YouTube. <https://www.youtube.com/watch?v=Y9xA3itXQes>

OpenAI. (2025a, 25 de mayo). *¿Cómo ponerte de coautor en una presentación tecnológica?* [ChatGPT]. <https://chatgpt.com/share/68334cc8-cfdc-8013-a361-9081ad98f20d>

OpenAI. (2025b). *Respuesta generada por ChatGPT sobre programación por lotes en Windows y Linux* [ChatGPT conversación]. <https://chatgpt.com/share/683354cd-e890-8013-a3a7-df827a7cf6eb>

MSYS2. (s.f.). *MSYS2 - Open Collective*. Open Collective. <https://opencollective.com/msys2>

BIBLIOGRAFÍA

Kernighan, B. W. y Ritchie, D. M. (1988). *El Lenguaje de Programación C* (2.^a ed., Versión en español). Trad. disponible en InfoBiblioteca Digital. Recuperado de <https://info-biblioteca.mincyt.gob.ve/books/el-lenguaje-de-programacion-c-espanol/>

Bjarne Stroustrup (2024). *Programming -- Principles and Practice Using C++* (3rd Edition). <https://www.stroustrup.com/programming.html>

Canva (s.f.). *Diseño de Carátula*. <https://www.canva.com>

Howard, C (2013, Oct. 5). *F-35 Joint Strike Fighter benefits from modern software testing, quality assurance*. Military-Aerospace Electronics. <https://www.militaryaerospace.com/home/article/16715746/f-35-joint-strike-fighter-benefits-from-modern-software-testing-quality-assurance>

open62541 Project. (s.f.). *open62541 – Open Source OPC UA (IEC62541) implementation*. <https://www.open62541.org/>

libmodbus Project. (s.f.). *libmodbus – A Modbus library for Linux, Mac OS X, FreeBSD, QNX and Windows*. <https://libmodbus.org/>

FreeCAD Project. (s.f.). *FreeCAD: Your own 3D parametric modeler*. <https://www.freecad.org/>

FreeCAD Project. (s.f.a). *FreeCAD (Version en desarrollo) [Código fuente]*. GitHub. <https://github.com/FreeCAD/FreeCAD>

OpenAI. (2025c). *Optimización de Windows 10 Pro en VirtualBox para fines académicos en un sistema Linux host [Asistencia de modelo de lenguaje]*. ChatGPT. <https://chatgpt.com/share/6833c311-cc34-8013-822c-30d3546789a7>