



Universidad Nacional Autónoma de México
Facultad de Ingeniería
División de Ingeniería Eléctrica
Ingeniería en Computación
Estructura y Programación de Computadoras



Semestre: 2021-I

Grupo: 2

Manual de configuración de software en Linux (Ubuntu)

Objetivo:

Configurar el entorno de desarrollo para el curso de Estructura y Programación de Computadoras en Sistema Operativo GNU/Linux para programación en lenguaje ensamblador de arquitectura Intel x86.

Requisitos:

- Sistema Operativo GNU/Linux (basado en Debian)
- Archivo Tasm.zip (compartido a través de Google Classroom)

Nota:

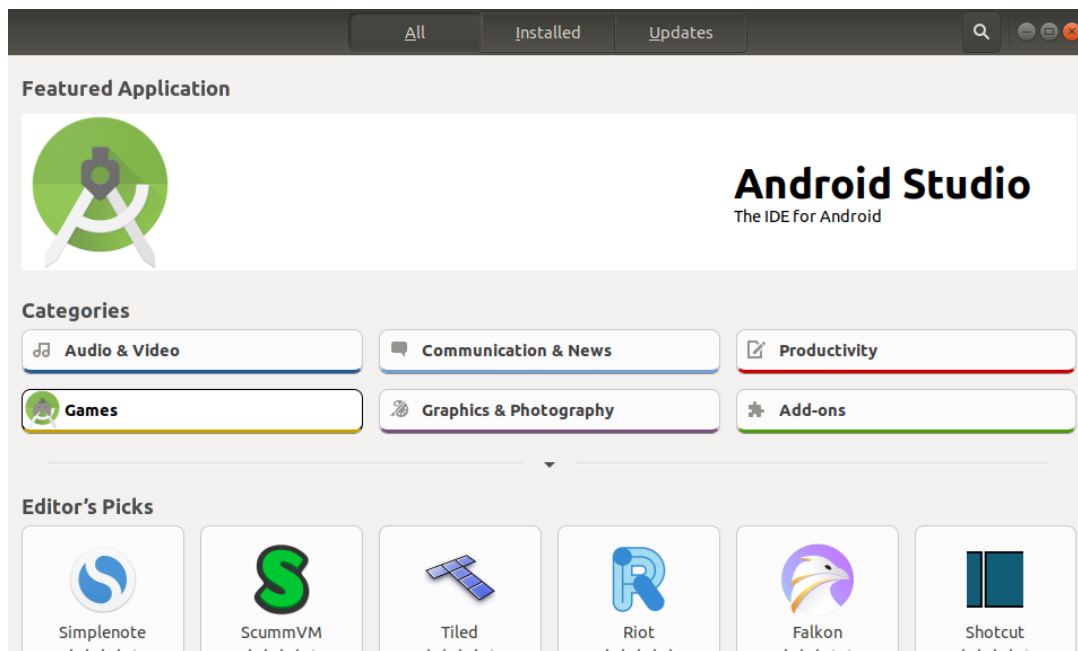
Este manual está desarrollado para hacer la instalación del programa DOSBox en un S.O. Ubuntu 18.04 LTS, y podría ser compatible a partir de la versión 16.04. Si se utiliza una distribución diferente, que no esté basada en Debian, los pasos pueden cambiar, si es el caso, consulte en internet los pasos necesarios para instalar DOSBox en su S.O.

Procedimiento:

1. Para instalar DOSBox se pueden utilizar los siguientes dos métodos:

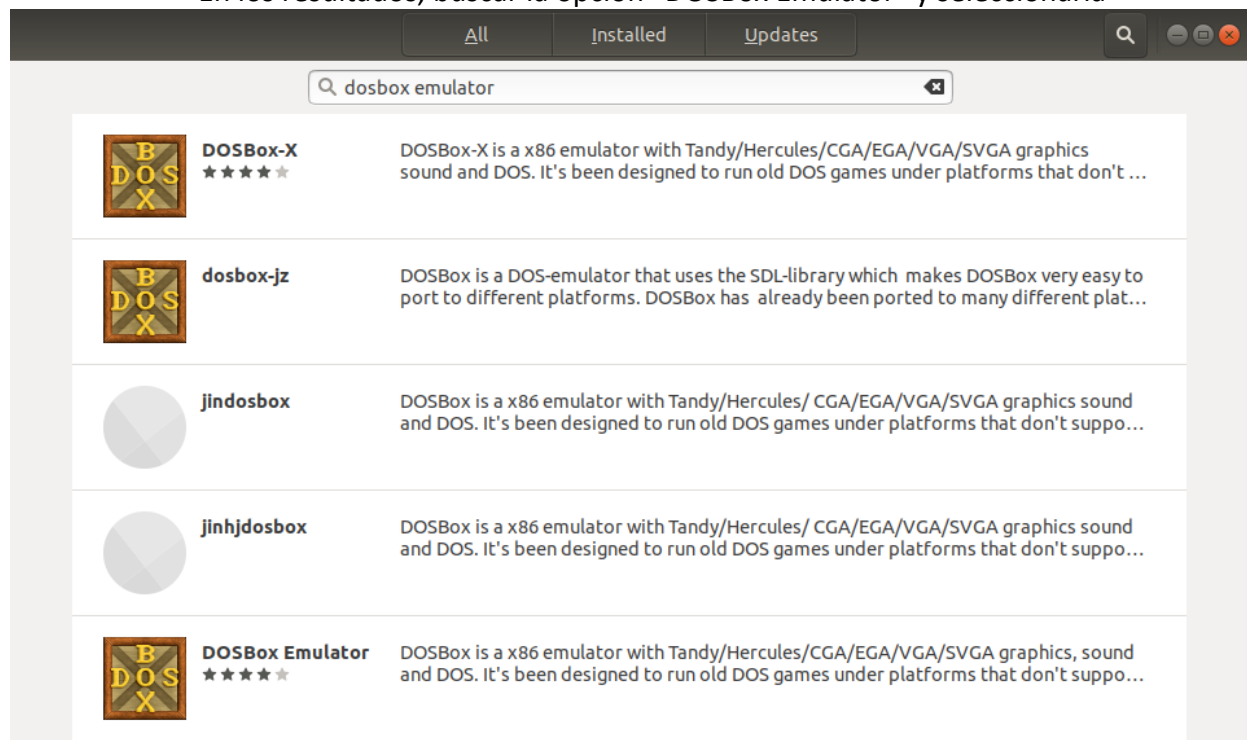
Utilizando el gestor de aplicaciones de Ubuntu (Ubuntu Software)

- a. El primer método es instalar desde el gestor de aplicaciones de Ubuntu, llamado Ubuntu Software (sólo en distribuciones de Ubuntu). Para esto, abrir el buscador de aplicaciones (tecla Windows) y teclear "Ubuntu software", dar [enter]. Se abrirá "Ubuntu Software" que luce de la siguiente manera:



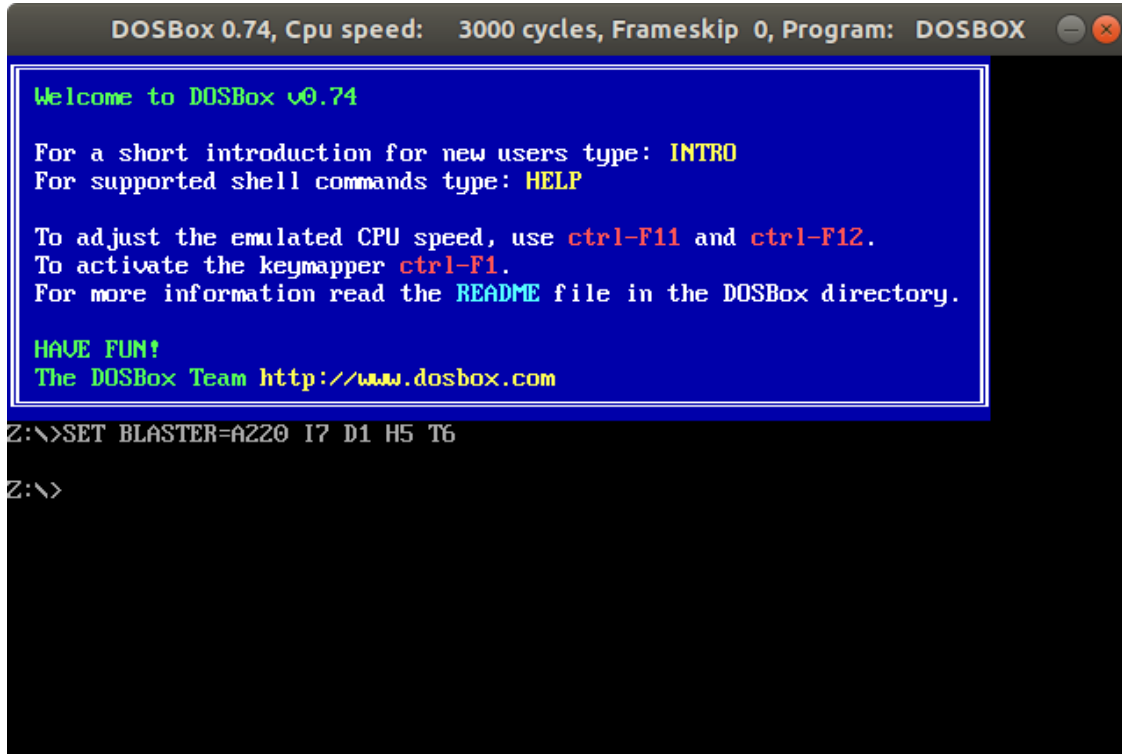
- b. Una vez abierto “Ubuntu Software”, dar clic izquierdo en el ícono de lupa (en la parte superior derecha) y teclear “dosbox emulator”, dar [enter].

En los resultados, buscar la opción “DOSBox Emulator” y seleccionarla



- c. Al seleccionar la aplicación se mostrará información como: descripción, capturas de pantalla, calificación, etc. Hacer clic en el botón ‘Instalar’ y la aplicación se instalará.

- d. Una vez instalada, hacer clic en el botón 'Ejecutar' y la aplicación se abrirá. La aplicación también puede abrirse desde el buscador de aplicaciones (presionar tecla Windows), tecleando "dosbox" y se mostrará la aplicación. Al abrir DOSBox, se abrirá una ventana nueva que muestra una aplicación con una terminal:



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

Welcome to DOSBox v0.74

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

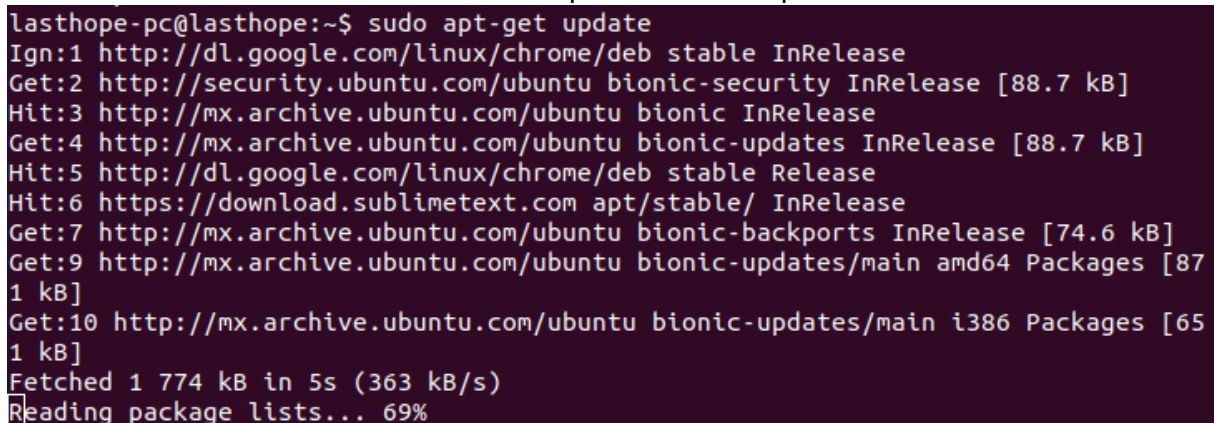
Z:\>
```

Utilizando el gestor de paquetes de Debian, *Advanced Packaging Tool* (APT), a través de la terminal.

- a. Abrir la terminal del sistema (en algunos sistemas se puede abrir con la combinación de teclas Ctrl+Shft+t).

- b. Teclear el comando:
\$sudo apt-get update

Este comando actualiza los repositorios a los que tiene acceso el sistema

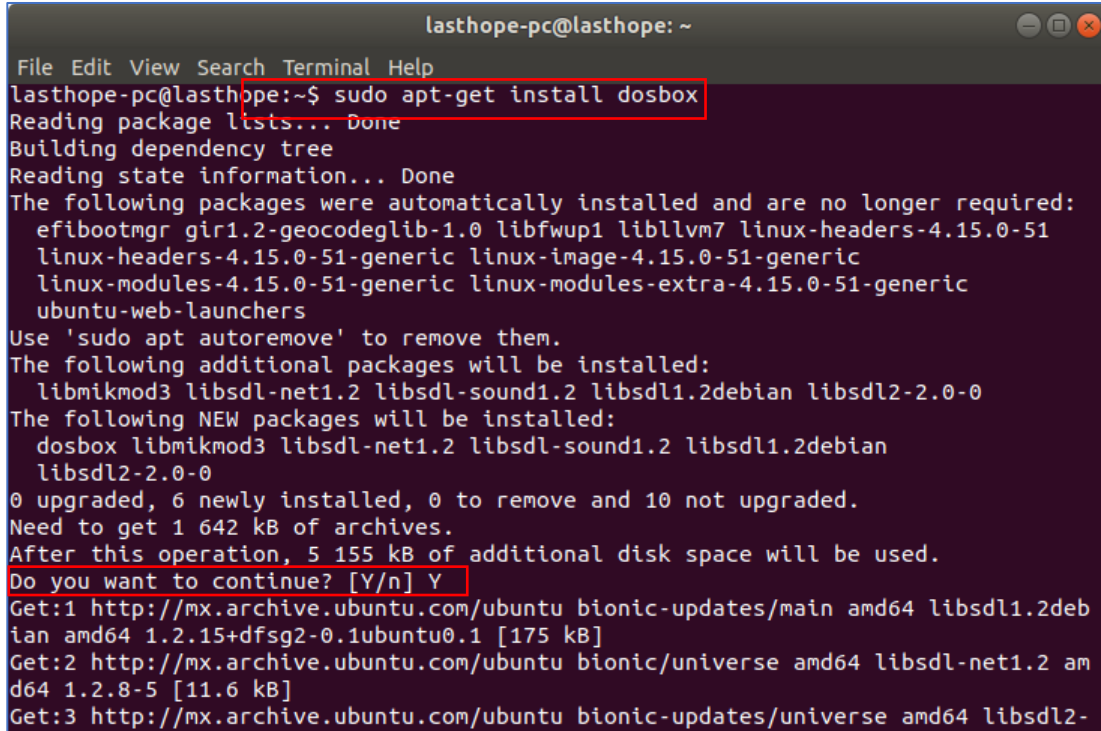


```
lasthope-pc@lasthope:~$ sudo apt-get update
Ign:1 http://dl.google.com/linux/chrome/deb stable InRelease
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Hit:3 http://mx.archive.ubuntu.com/ubuntu bionic InRelease
Get:4 http://mx.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Hit:5 http://dl.google.com/linux/chrome/deb stable Release
Hit:6 https://download.sublimetext.com apt/stable/ InRelease
Get:7 http://mx.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:9 http://mx.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [87
1 kB]
Get:10 http://mx.archive.ubuntu.com/ubuntu bionic-updates/main i386 Packages [65
1 kB]
Fetched 1 774 kB in 5s (363 kB/s)
Reading package lists... 69%
```

- c. Teclear el comando:

\$sudo apt-get install dosbox

Durante el proceso de instalación en la terminal, el sistema pregunta si se desean instalar algunos paquetes adicionales (dependencias). Teclear 'y' y dar [enter] para instalarlas.



```
lasthope-pc@lasthope: ~  
File Edit View Search Terminal Help  
lasthope-pc@lasthope:~$ sudo apt-get install dosbox  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  efibootmgr gir1.2-geocodeglib-1.0 libfwup1 libllvm7 linux-headers-4.15.0-51  
  linux-headers-4.15.0-51-generic linux-image-4.15.0-51-generic  
  linux-modules-4.15.0-51-generic linux-modules-extra-4.15.0-51-generic  
  ubuntu-web-launchers  
Use 'sudo apt autoremove' to remove them.  
The following additional packages will be installed:  
  libmikmod3 libsdl-net1.2 libsdl-sound1.2 libsdl1.2debian libsdl2-2.0-0  
The following NEW packages will be installed:  
  dosbox libmikmod3 libsdl-net1.2 libsdl-sound1.2 libsdl1.2debian  
  libsdl2-2.0-0  
0 upgraded, 6 newly installed, 0 to remove and 10 not upgraded.  
Need to get 1 642 kB of archives.  
After this operation, 5 155 kB of additional disk space will be used.  
Do you want to continue? [Y/n] Y  
Get:1 http://mx.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libsdl1.2deb  
ian amd64 1.2.15+dfsg2-0.1ubuntu0.1 [175 kB]  
Get:2 http://mx.archive.ubuntu.com/ubuntu bionic/universe amd64 libsdl-net1.2 am  
d64 1.2.8-5 [11.6 kB]  
Get:3 http://mx.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 libsdl2-
```

- d. Al finalizar el proceso de instalación, y si no ocurre ningún inconveniente, DOSBox estará instalado. Para abrirlo, se debe hacer desde la terminal tecleando el comando:

\$dosbox

Se abrirá una ventana nueva en donde se ejecutará el emulador.

```
lasthope-pc@lasthope: ~  
File Edit View Search Terminal Help  
lasthope-pc@lasthope:~$ dosbox  
DOSBox version 0.74  
Copyright 2002-2010 DOSBox Team, published under GNU GPL.  
---  
CONFIG: Generating default configuration.  
Writing it to /home/lasthope-pc/.dosbox/dosbox-0.74.conf  
CONFIG:Loading primary settings from config file /home/lasthope-pc/.dosbox/dosbox-0.74.conf  
MIXER:Got different values from SDL: freq 44100, blocksize 512  
ALSA lib conf.c:3314:(snd_config_hooks_call) Cannot open shared library libasound_module_conf_pulse.so  
ALSA lib seq.c:935:(snd_seq_open_noupdate) Unknown SEQ hw  
ALSA:Can't open sequencer  
MIDI:Opened device:none  
lasthope-pc@lasthope:~$ dosbox  
DOSBox version 0.74  
Copyright 2002-2010 DOSBox Team, published under GNU GPL.  
---  
CONFIG:Loading primary settings from config file /home/lasthope-pc/.dosbox/dosbox-0.74.conf  
MIXER:Got different values from SDL: freq 44100, blocksize 512  
ALSA lib conf.c:3314:(snd_config_hooks_call) Cannot open shared library libasound_module_conf_pulse.so  
ALSA lib seq.c:935:(snd_seq_open_noupdate) Unknown SEQ hw
```

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX  
Welcome to DOSBox v0.74  
For a short introduction for new users type: INTRO  
For supported shell commands type: HELP  
To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.  
To activate the keymapper ctrl-F1.  
For more information read the README file in the DOSBox directory.  
HAVE FUN!  
The DOSBox Team http://www.dosbox.com  
Z:\>SET BLASTER=A220 I7 D1 H5 T6  
Z:\>
```

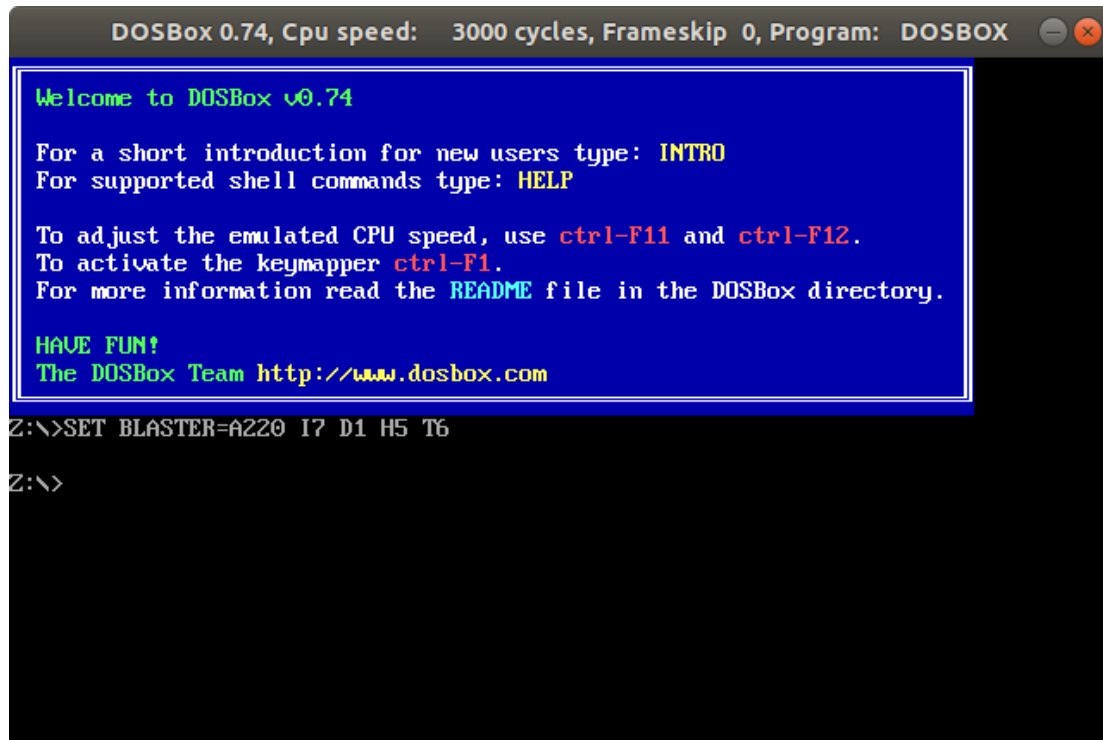
2. Descargar el archivo Tasm.zip desde la carpeta compartida de Google Drive.

3. Estando en Linux, descomprimir el contenido del archivo Tasm.zip en la carpeta Documentos (*Documents*) en la carpeta de *home* (/home/[USUARIO]/Documents).

Los archivos que se encuentran en la carpeta comprimida son los que se utilizan para poder ejecutar Turbo Assembler (TASM). Principalmente utilizaremos 3: tasm.exe (Turbo Assembler), tlink.exe (Turbo Linker) y td.exe (Turbo Debugger); sin embargo, los demás archivos son útiles para las herramientas que se proporcionan por el Turbo Debugger, como las opciones de Ayuda.

4. Abrir la aplicación DOSBox (omitir este paso si ya se encuentra abierta).
 - a. Si la aplicación se instaló desde la terminal, es necesario abrir una terminal y ejecutar el comando **\$dosbox**
 - b. Si la aplicación se instaló desde Ubuntu Software, abrir el buscador de aplicaciones y buscar 'dosbox'.

La ventana de DOSBox luce de la siguiente manera:



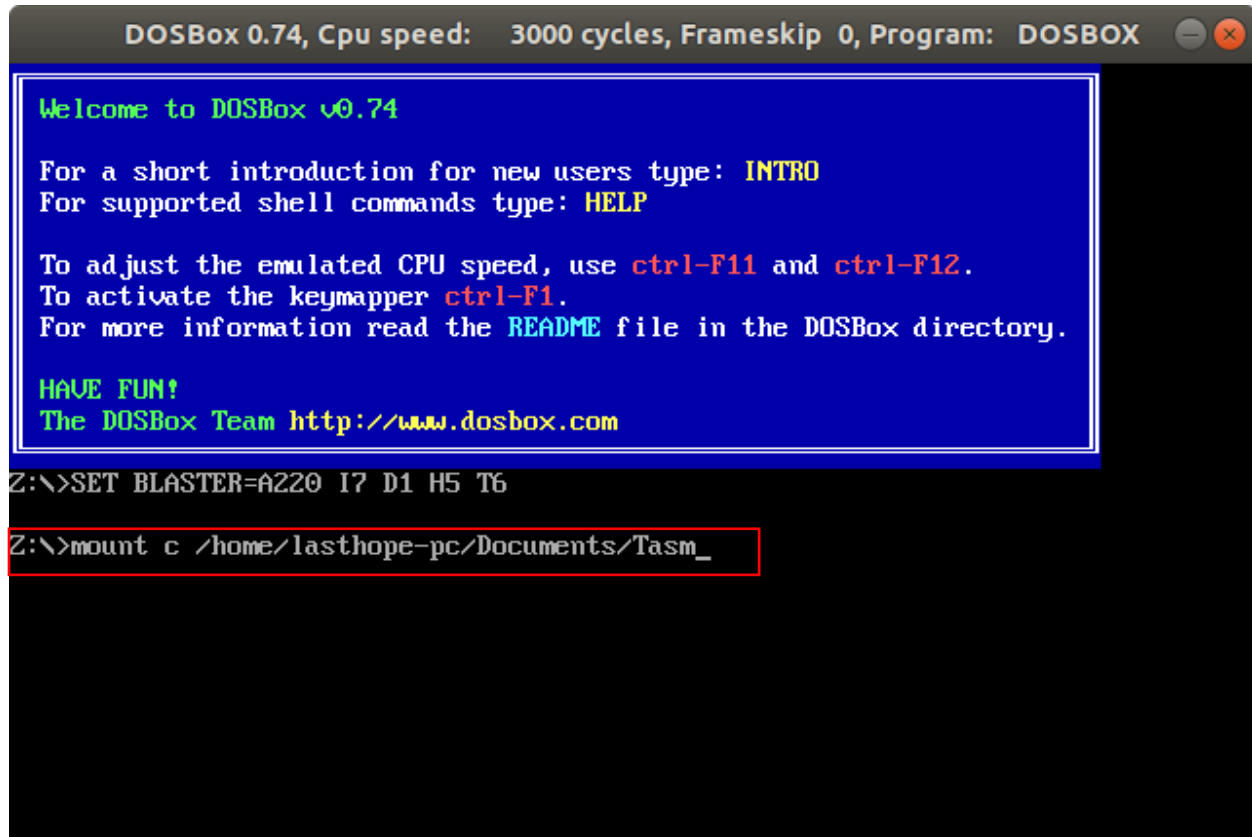
Esta terminal funciona del mismo modo que DOS. Inicialmente se ejecuta en una unidad distinta a la del disco duro. DOSBox se ejecuta en la unidad 'Z' por default. Para que se puedan ver los archivos de la computadora es necesario "montar" la unidad.

5. A continuación, sólo se montará una parte de la unidad del disco local. Se montará la carpeta de los archivos de Tasm, es decir, la carpeta que se encuentra en /home/[USUARIO]/Documents/Tasm
Obviamente, sustituir [USUARIO] por el nombre de usuario utilizado en la cuenta del sistema operativo.

Para lograr esto, en la terminal de DOSBox se introduce el comando:

```
Z:\> mount c /home/[USUARIO]/Documents/Tasm
```

Y dar enter.



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

Welcome to DOSBox v0.74

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount c /home/lasthope-pc/Documents/Tasm_
```

Con el comando anterior, se indica que se desea montar la carpeta /home/[USUARIO]/Documents/Tasm en una unidad virtual etiquetada con la letra C. De ese modo, se podrá acceder a la carpeta a través de la unidad virtual.

6. Para hacer el cambio de unidad, se introduce el siguiente comando:

```
Z:\> c:
```

Y dar enter

De esa manera, el prompt cambia a C:\> y eso indica que DOSBox ya se encuentra en carpeta /home/[USUARIO]/Documents/Tasm.

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

Welcome to DOSBox v0.74

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount c /home/lasthope-pc/Documents/Tasm
Drive C is mounted as local directory /home/lasthope-pc/Documents/Tasm/

Z:\>c:
```

7. Para confirmar que se encuentra funcionando, dentro de la carpeta comprimida Tasm hay una carpeta más, llamada 'eypc' en donde se encuentra un código fuente de ejemplo. El procedimiento para ensamblar un código fuente en lenguaje ensamblador es el siguiente:

- a. Desde la unidad C:, introducir el siguiente comando:

C:\> tasm eyec\ejemplo.asm

Y dar [enter].

Lo que sucede es que DOSBox busca el comando tasm dentro de la carpeta /home/[USUARIO]/Documents/Tasm y encuentra el archivo tasm.exe el cual será ejecutado. TASM ensambla el código fuente enviado como parámetro. En la terminal se observa el resultado del ensamblado y si no hubo errores, se genera un archivo objeto (.obj) por default en la misma ubicación que el archivo tasm.exe.


```
C:\>tasm eypc\ejemplo.asm
Turbo Assembler Version 4.1 Copyright (c) 1988, 1996 Borland International

Assembling file:  eypc\ejemplo.asm  to  ejemplo.OBJ
Error messages:   None
Warning messages: None
Passes:           1
Remaining memory: 466k

C:\>
```

- b. Ahora, desde la misma ubicación, introducir el comando:

C:\> tlink ejemplo.obj

Y dar [enter].

El comando tlink ejecuta el archivo tlink.exe, el cual es un enlazador y que va a generar un archivo ejecutable a partir del archivo objeto. Hace uso de uno o más archivos objeto de ser necesario. Al final se generará un archivo .exe y un archivo .map. Se ignorará este último.

```
C:\>tasm eypc\ejemplo.asm
Turbo Assembler Version 4.1 Copyright (c) 1988, 1996 Borland International

Assembling file:  eypc\ejemplo.asm  to  ejemplo.OBJ
Error messages:   None
Warning messages: None
Passes:          1
Remaining memory: 466k

C:\>tlink ejemplo.obj
Turbo Link Version 7.1.30.1. Copyright (c) 1987, 1996 Borland International

C:\>_
```

- c. Para finalizar, se introducirá el siguiente comando:

C:\> td ejemplo.exe

Y dar [enter].

El comando td ejecuta el archivo td.exe, el Turbo Debugger, que se usará para hacer la depuración del programa generado. Aquí será posible revisar la ejecución el código línea por línea para corroborar su correcto funcionamiento.

```
C:\>tasm eypc\ejemplo.asm
Turbo Assembler Version 4.1 Copyright (c) 1988, 1996 Borland International

Assembling file:   eypc\ejemplo.asm to ejemplo.OBJ
Error messages:   None
Warning messages: None
Passes:           1
Remaining memory: 466k

C:\>tlink ejemplo.obj
Turbo Link Version 7.1.30.1. Copyright (c) 1987, 1996 Borland International

C:\>td ejemplo.exe
```

- d. El comando anterior abre el *Turbo Debugger* en el cual es posible depurar el código en lenguaje ensamblador. Para más detalles al respecto, ver Anexo 1 – *Turbo Debugger*.
8. Para editar el código fuente en lenguaje ensamblador se utilizará un programa editor de texto que es robusto y muy socorrido por desarrolladores en el campo laboral. Este programa se llama Sublime Text y se puede ver más información al respecto en el Anexo 2 – Editor de Texto *Sublime Text*

Anexo 1 – Turbo Debugger

Turbo Debugger es un programa utilizado para realizar la depuración de archivos ejecutables en DOS. Este programa es útil para depurar los programas que se generarán a partir del lenguaje ensamblador para arquitectura x86.

La ventana de *Turbo Debugger* es la siguiente:

The screenshot displays the Turbo Debugger window with the following components:

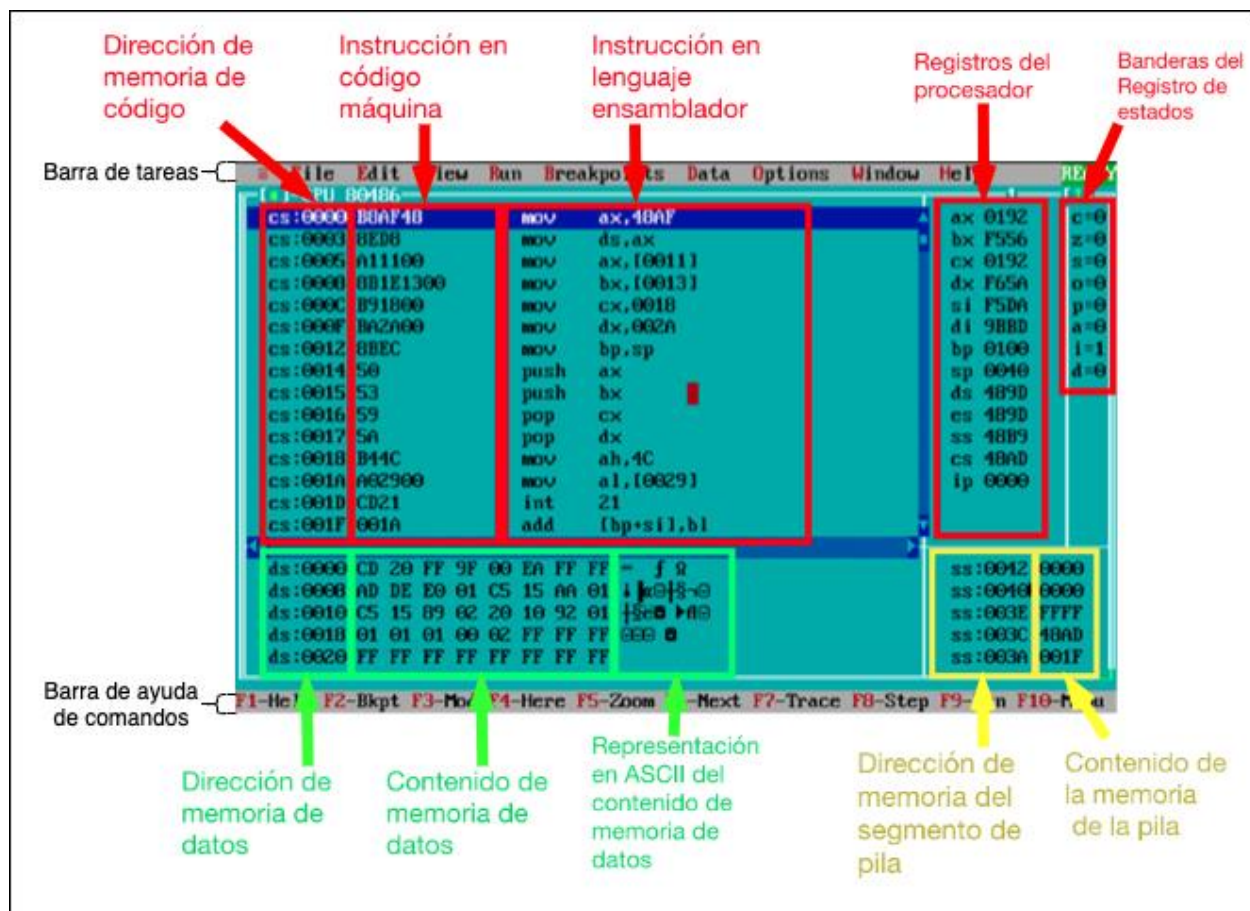
- Menu Bar:** File, Edit, View, Run, Breakpoints, Data, Options, Window, Help.
- Status Bar:** CPU 80486, READY.
- Assembly Window:** Displays assembly instructions with their addresses and hex values.

Address	Hex	Instruction
cs:0000	B8AF48	mov ax,48AF
cs:0003	8ED8	mov ds,ax
cs:0005	A11100	mov ax,[0011]
cs:0008	BB1E1300	mov bx,[0013]
cs:000C	B91800	mov cx,0018
cs:000F	BA2A00	mov dx,002A
cs:0012	8BEC	mov bp,sp
cs:0014	50	push ax
cs:0015	53	push bx
cs:0016	59	pop cx
cs:0017	5A	pop dx
cs:0018	B44C	mov ah,4C
cs:001A	A02900	mov al,[0029]
cs:001D	CD21	int 21
cs:001F	001A	add [bp+si],bl
- Registers Window:** Shows the current values of registers.

Register	Value
ax	0192
bx	F556
cx	0192
dx	F65A
si	F5DA
di	98BD
bp	0100
sp	0040
ds	489D
es	489D
ss	48B9
cs	48AD
ip	0000
- Memory Window:** Displays memory contents in hex and ASCII.

Address	Hex	ASCII
ds:0000	CD 20 FF 9F 00 EA FF FF	- f n
ds:0008	AD DE E0 01 C5 15 AA 01	↓ k 0 \$ - 0
ds:0010	C5 15 89 02 20 10 92 01	\$ 0 0 > A 0
ds:0018	01 01 01 00 02 FF FF FF	000 0
ds:0020	FF FF FF FF FF FF FF	
- Stack Window:** Shows stack memory.

Address	Value
ss:0042	0000
ss:0040	0000
ss:003E	FFFF
ss:003C	48AD
ss:003A	001F
- Footer:** F1-Help F2-Bkpt F3-Mod F4-Here F5-Zoom F6-Next F7-Trace F8-Step F9-Run F10-Menu



NOTA:

Inicialmente, el registro de segmento de datos (DS) apunta a un segmento aleatorio que no necesariamente es el segmento reservado para el programa. Cuando se hace uso del segmento de datos en un programa, siempre es necesario ejecutar 2 instrucciones al inicio para que el programa sepa cuál es el segmento correcto al que debe apuntar el registro DS. Estas dos instrucciones son:

```
mov ax, @data
```

```
mov ds, ax
```

@data es una palabra reservada por el ensamblador TASM que le indicará al programa la dirección del segmento de datos. La primera instrucción servirá para obtener esa dirección y ponerla en el registro AX, de manera que la segunda instrucción permite inicializar el registro DS con ese valor. Recordar que al registro DS no se puede asignar un valor inmediato, es por eso que se hace uso del registro AX como intermediario. Después de esas dos instrucciones.

Durante la depuración, el código se puede ejecutar línea por línea o ejecutar un bloque de código estableciendo puntos de quiebre (*breakpoints*) para detener la ejecución en donde éste se encuentre.

En la sección del segmento de datos, se puede observar el contenido de la memoria (incluyendo memoria de programa, datos y pila). Para cambiar la localidad que se visualiza, es posible dar clic derecho sobre la sección, y seleccionar la opción "Goto...". Entonces preguntará

por la dirección de la localidad, se puede introducir una dirección directamente: "48FD:0000", o se pueden utilizar los registros de segmento como apoyo: "ds:0000", luego seleccionar "Ok".

También es posible modificar el contenido de la memoria byte por byte directamente en la sección de datos. Sólo es necesario dar clic sobre el byte a modificar e introducir el valor, inmediatamente se abrirá un cuadro de texto mostrando el valor que se está introduciendo. Una vez que se finalizó la entrada de texto, hacer clic para cambiar el valor. Hay un detalle con valores a partir de A0 a FF ya que no es posible reconocer dichos valores si no se pone un '0' a la izquierda de éstos. Es decir, para introducir el valor 'B5' en una localidad, se deberá introducir el valor '0B5' en el cuadro de texto.

Anexo 2 – Editor de Texto *Sublime Text*

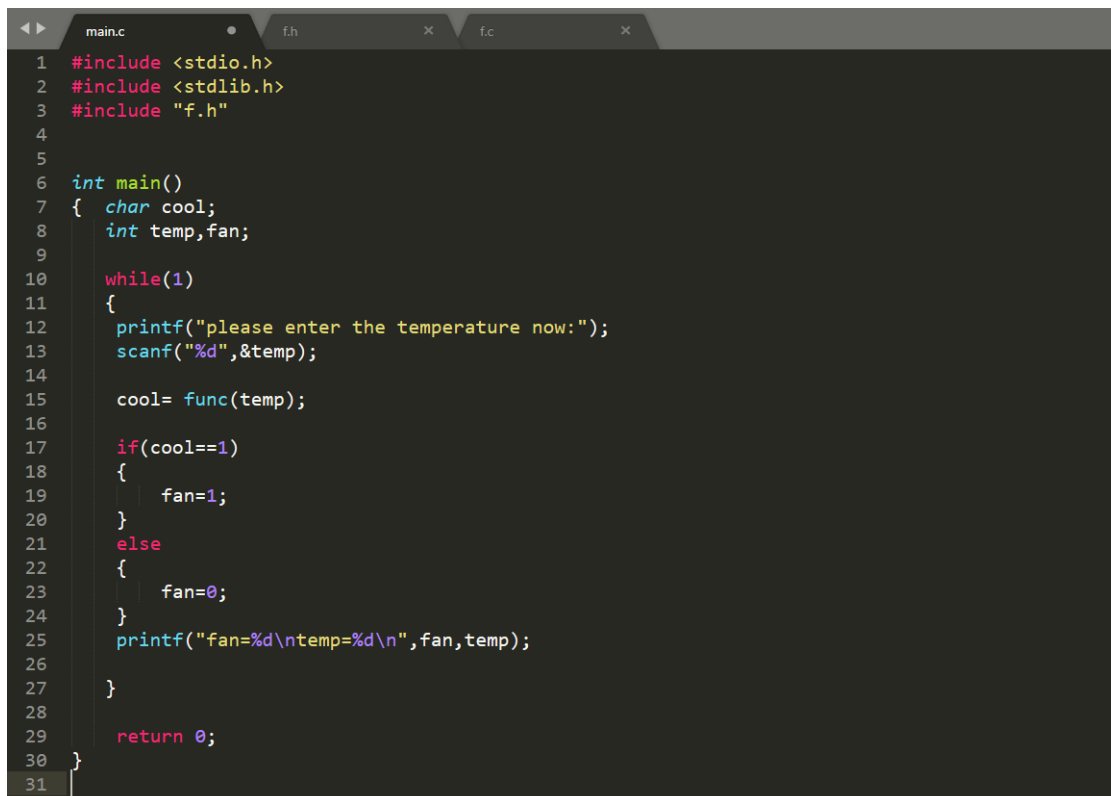
Para crear un programa en lenguaje ensamblador se puede utilizar cualquier editor de texto plano. Cada archivo en lenguaje ensamblador deberá guardarse con extensión .asm.

Los editores de texto más comunes son: Bloc de notas o Notepad++, en Windows; Gedit, Nano, o Vi, en Linux; o TextEdit en MacOS. Sin embargo, es posible que la escritura de código en alguno de éstos se vuelva complicada debido a que es difícil editar un archivo de texto plano sin formato. De esta manera es fácil cometer un error.

Existe un editor de texto útil para desarrolladores que permite reconocer diferentes extensiones de archivos para ponerles formato de acuerdo con la sintaxis del lenguaje de programación utilizado. Este editor de texto se llama *Sublime Text* y se puede encontrar tanto para Windows, como para Linux y MacOS. <https://www.sublimetext.com/>

De manera predeterminada, Sublime Text tiene herramientas que reconocen más de 30 lenguajes de programación, como C, C++, C#, Java, y Python, incluso HTML. Pero una característica importante de Sublime Text es que permite agregar paquetes (*packages*) que permiten agregar aún más lenguajes o funciones al editor de texto, haciéndolo más robusto.

Por default, Sublime Text no contiene herramientas para reconocer la sintaxis del lenguaje ensamblador para arquitectura x86, aunque existe un *package* que lo permite. Se recomienda utilizar este editor de texto durante el curso.



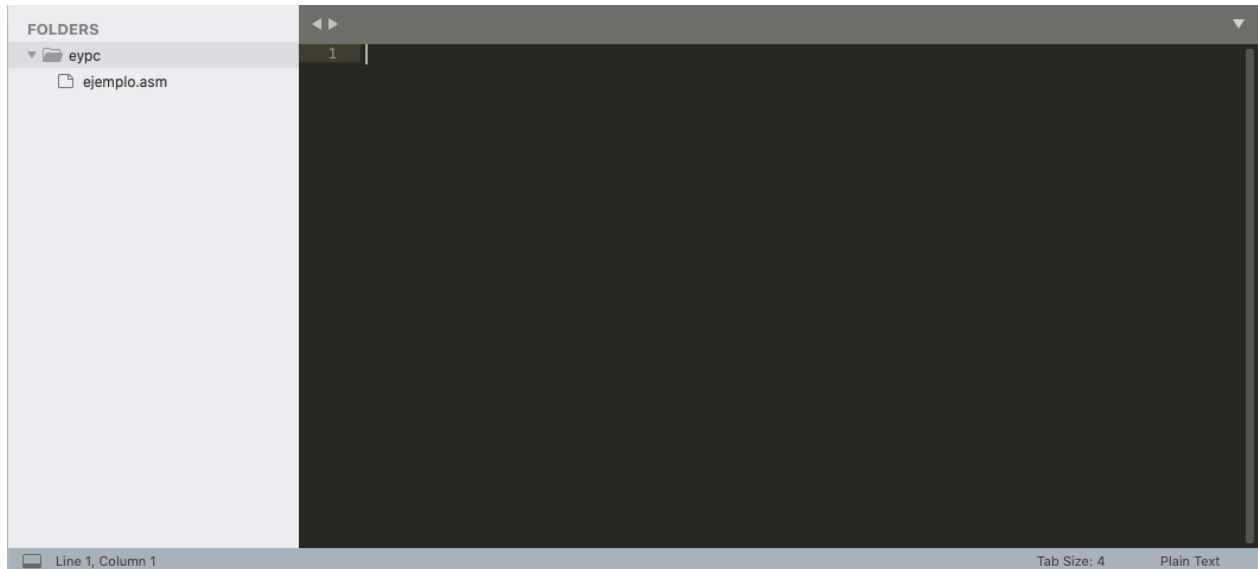
```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include "f.h"
4
5
6 int main()
7 { char cool;
8   int temp,fan;
9
10  while(1)
11  {
12    printf("please enter the temperature now:");
13    scanf("%d",&temp);
14
15    cool= func(temp);
16
17    if(cool==1)
18    {
19      fan=1;
20    }
21    else
22    {
23      fan=0;
24    }
25    printf("fan=%d\ntemp=%d\n",fan,temp);
26
27  }
28
29  return 0;
30 }
```

Para instalar Sublime Text en Ubuntu 18.04, seguir los pasos del siguiente tutorial:

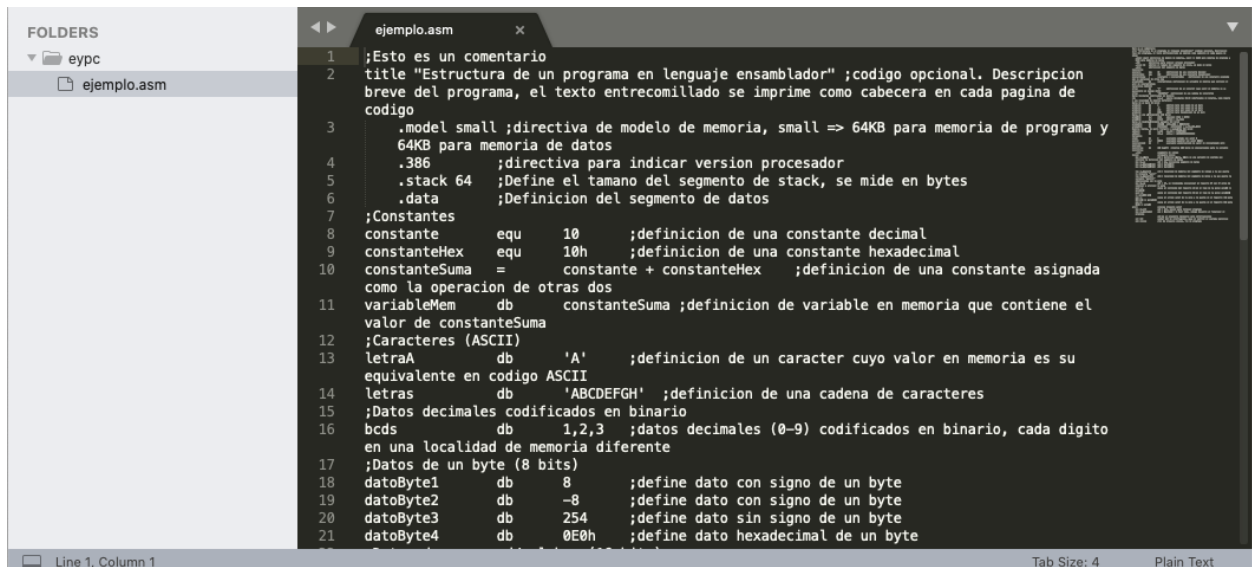
<https://linuxconfig.org/how-to-install-sublime-text-on-ubuntu-18-04-bionic-beaver-linux>

Para otros sistemas operativos que no estén basados en Debian, consultar en internet.

Una vez instalado, abrir el editor de texto. Dentro del editor de texto, en la barra de tareas, ir a File -> Open... (Archivo -> Abrir...) y seleccionar la carpeta 'eypc' dentro de '/home/[USUARIO]/Documents/Tasm' y esto mostrará los archivos en esa carpeta del lado izquierdo.

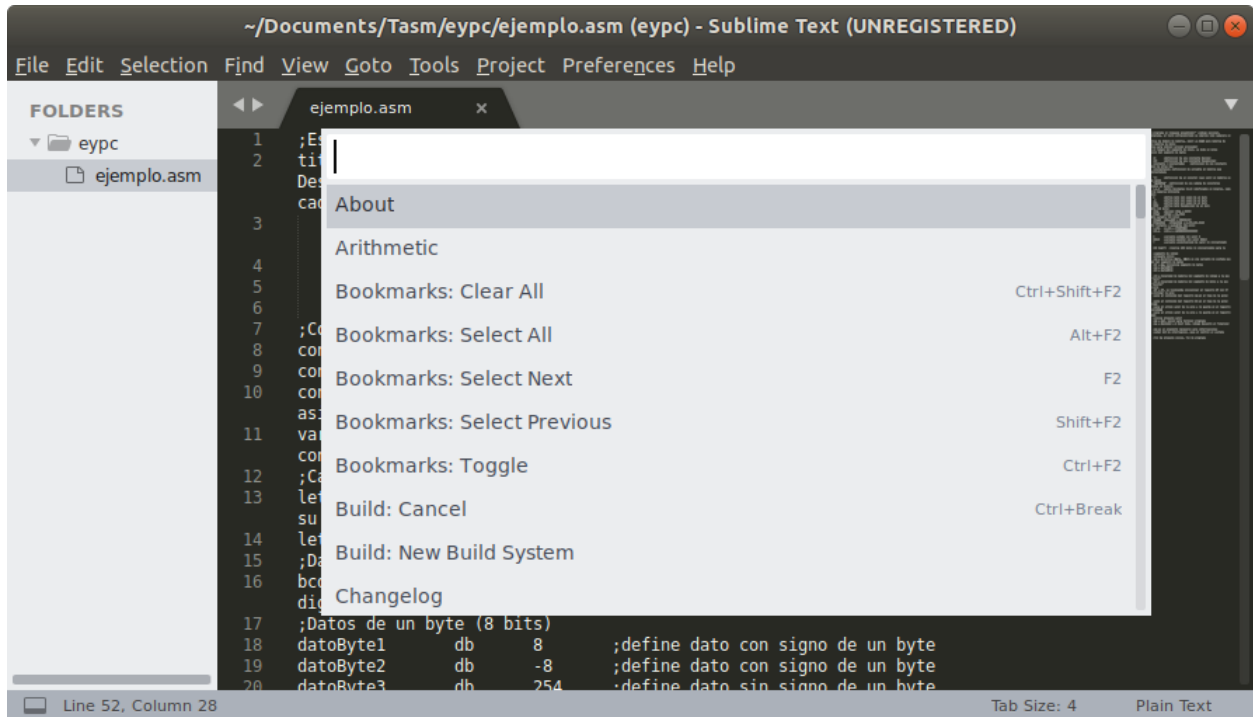


Si se abre el archivo ejemplo.asm, éste se mostrará como de texto plano, dado que aún no se instala el paquete necesario para reconocer la sintaxis del lenguaje ensamblador arquitectura x86.

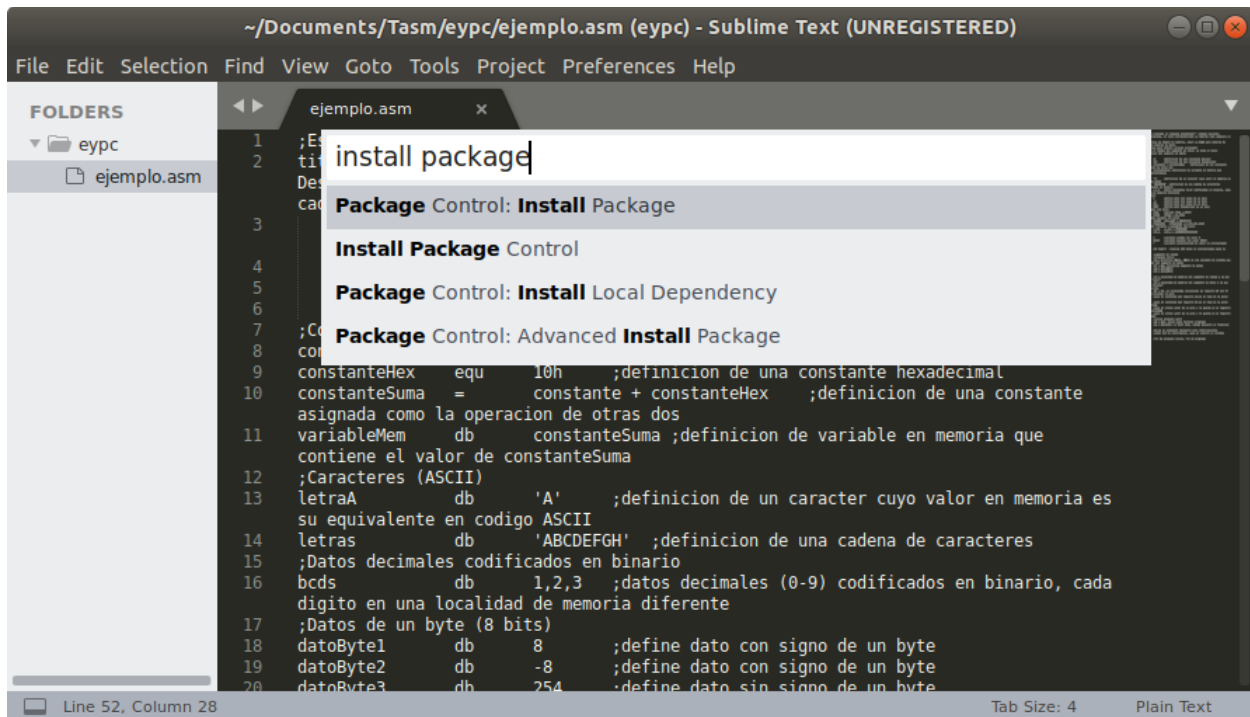


Para instalar el paquete que reconoce la sintaxis del x86, en primer lugar se debe actualizar algo llamado *Package Control*. *Package Control* es el gestor que permite instalar paquetes dentro de Sublime Text. Para actualizarlo, ir a la opción **Tools** (Herramientas) en la barra de tareas y seleccionar *Install Package Control* (Instalar Package Control) del menú.

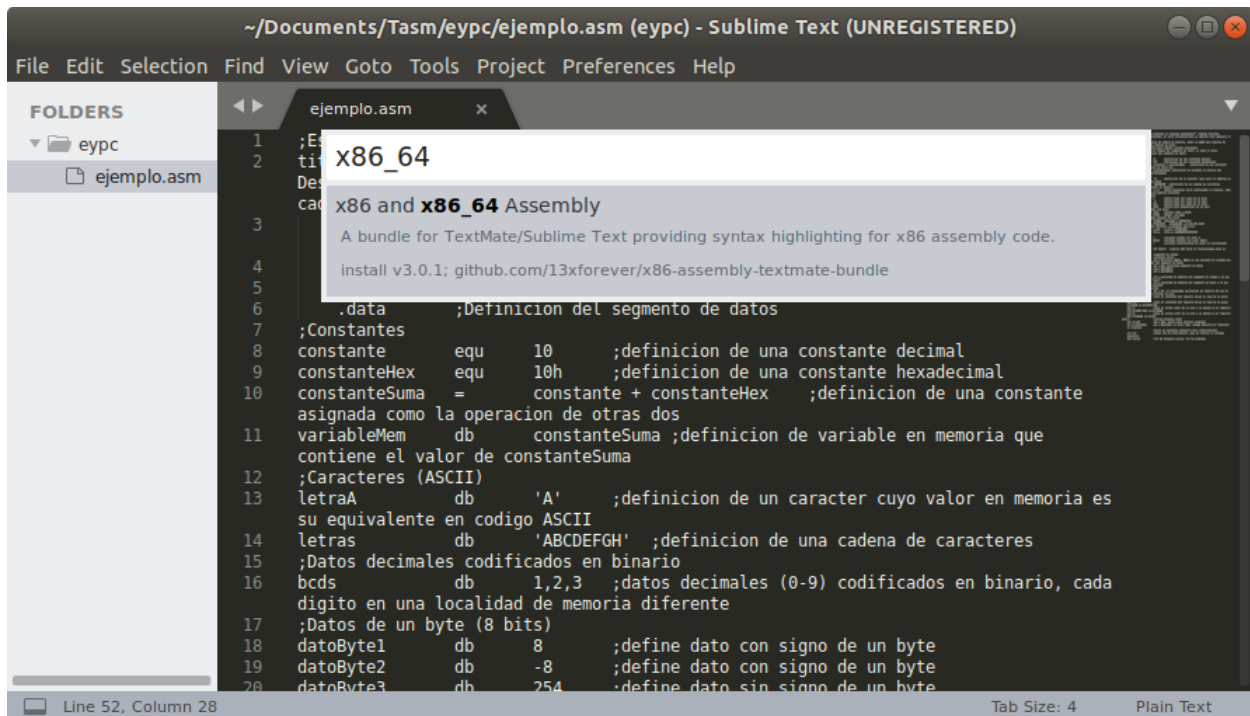
Una vez actualizado, para instalar un paquete se ir a **Tools** (Herramientas) en la barra de tareas y seleccionar *Command Palette...* (Paleta de Comandos...). Se mostrará un campo de texto y una lista de comandos.



Teclear “install package” (o “instalar paquete”, si está en español) y seleccionar “*Package Control: Install Package*” (“Package Control: Instalar Paquete”).

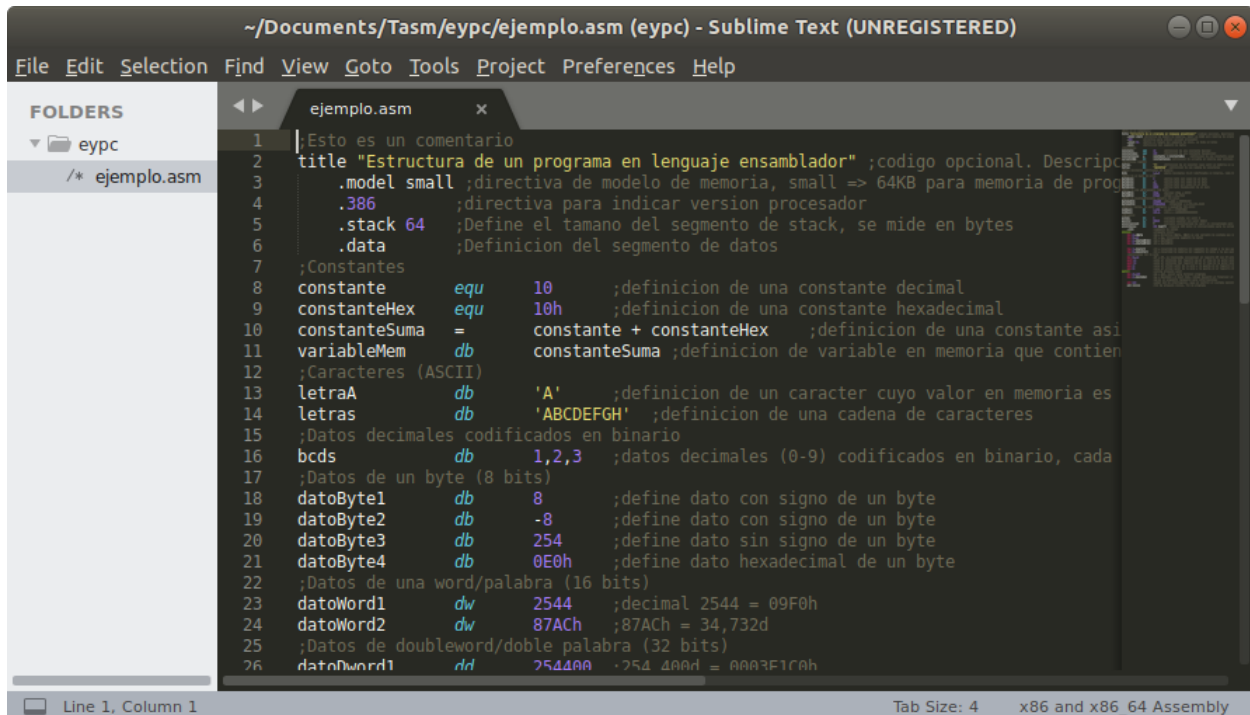


Se abre una nueva lista, teclear “x86_64” y buscar la opción “x86 and x86_64 Assembly”.
Seleccionar tal paquete y en ese momento comenzará a instalarlo.



En la parte baja del editor se muestra la operación. Cuando termine la instalación del paquete, cerrar la pestaña del archivo y volver a abrirlo, entonces ahora se deberá mostrar el texto con formato y estilo que reconoce por defecto la sintaxis de la arquitectura x86.

Es posible observar en la esquina inferior derecha la sintaxis del lenguaje que se reconoció. Si se detecta que el archivo es de lenguaje ensamblador, entonces por defecto el estilo seleccionado será el del "x86 and x86_64 Assembly"



```
1 ;Esto es un comentario
2 title "Estructura de un programa en lenguaje ensamblador" ;codigo opcional. Descripc
3 .model small ;directiva de modelo de memoria, small => 64KB para memoria de proc
4 .386 ;directiva para indicar version procesador
5 .stack 64 ;Define el tamaño del segmento de stack, se mide en bytes
6 .data ;Definición del segmento de datos
7 ;Constantes
8 constante equ 10 ;definición de una constante decimal
9 constanteHex equ 10h ;definición de una constante hexadecimal
10 constanteSuma = constante + constanteHex ;definición de una constante asi
11 variableMem db constanteSuma ;definición de variable en memoria que contien
12 ;Caracteres (ASCII)
13 letraA db 'A' ;definición de un caracter cuyo valor en memoria es
14 letras db 'ABCDEFGH' ;definición de una cadena de caracteres
15 ;Datos decimales codificados en binario
16 bcdd db 1,2,3 ;datos decimales (0-9) codificados en binario, cada
17 ;Datos de un byte (8 bits)
18 datoByte1 db 8 ;define dato con signo de un byte
19 datoByte2 db -8 ;define dato con signo de un byte
20 datoByte3 db 254 ;define dato sin signo de un byte
21 datoByte4 db 0E0h ;define dato hexadecimal de un byte
22 ;Datos de una word/palabra (16 bits)
23 datoWord1 dw 2544 ;decimal 2544 = 09F0h
24 datoWord2 dw 87ACh ;87ACh = 34,732d
25 ;Datos de doubleword/doble palabra (32 bits)
26 datoDword1 dd 2544AA ;2544 AAAA = AAA3F1CAh
```

Con Sublime Text se puede editar el texto y guardarlo sin ningún problema. Intente editar el texto y ensamblarlo de nueva cuenta en DOSBox para detectar los cambios en el código y ver cómo se ejecutan.