	Caratula para entrega de prácticas
Facultad de ingeniería	Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Alejandro Pimentel

Asignatura: Fundamentos de Programación

Grupo: 3

No de Práctica(s): 6

Integrante(s): Arteaga Munguía Erick Alejandro

*No. de Equipo de
cómputo empleado:* Namibia

No. de Lista o Brigada: 6294

Semestre: 2020-1

Fecha de entrega: 30/09/19

Observaciones: Era mucho más sencillo que descargaras el archivo de texto y que lo abrieras usando vim. de esa manera podías trabajar sobre el mismo archivo.

CALIFICACIÓN: 10

Objetivo:

Conocer y usar los ambientes y herramientas para el desarrollo y ejecución de programas en Lenguaje C, como editores y compiladores en diversos sistemas operativos.

¿Que es TXT?

Un archivo de texto simple, texto sencillo o texto sin formato, es un archivo informático que contiene únicamente texto formado solo por caracteres que son legibles por humanos, careciendo de cualquier tipo de formato tipográfico.

Estos archivos están compuestos de bytes que representan caracteres ordinarios como letras, números y signos de puntuación (incluyendo espacios en blanco), también incluye algunos pocos caracteres de control como tabulaciones, saltos de línea y retornos de carro. Estos caracteres se pueden codificar de distintos modos. El sistema de codificación ASCII viene a ser la base primordial y no necesita de un identificador explícito en la comunicación digital. Para poder representar diferentes sistemas de codificación de caracteres como UTF-1, UTF-7, UTF-8, UTF-16, UTF-32, UTF-EBCDIC, SCSU, BOCU-1 y GB 18030 es necesario que haya al principio de cada fichero de texto una marca de orden de bytes.

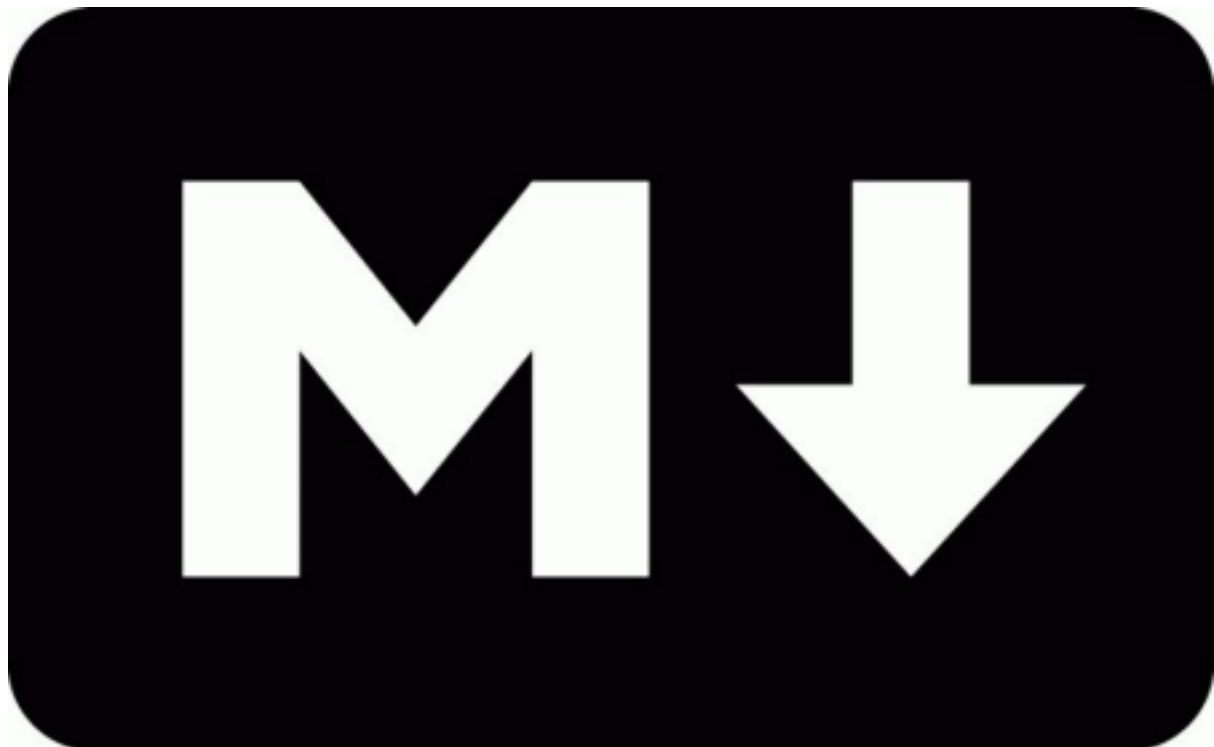
Los archivos de texto llano carecen de información destinada a generar formatos (negritas, subrayado, cursivas, tamaño, etc.) y tipos de letra (por ejemplo, Arial, Times, Courier, etc.). Esta simplicidad permite que una gran variedad de programas pueda leer y editar este contenido. Las aplicaciones destinadas a la escritura y modificación de archivos de texto se llaman editores de texto. Cada texto tiene un peso diferente porque tiene una codificación distinta.



¿Qué es markdown?

Markdown es un lenguaje de marcado que *facilita la aplicación de formato* a un texto empleando una serie de caracteres de una forma especial. En principio, fue pensado para elaborar textos cuyo destino iba a ser la web con más rapidez y sencillez que si estuviésemos empleando directamente HTML. Y si bien este suele ser el mejor uso que podemos darle, también podemos emplearlo para cualquier tipo de texto, independientemente de cual vaya a ser su destino.

Markdown es realmente dos cosas: por un lado, el lenguaje; por otro, una herramienta de software que convierte el lenguaje en HTML válido. Para entenderlo mejor, veámoslo con un ejemplo. Digamos que queremos escribir un encabezado de nivel 1. Esto en HTML se hace con la etiqueta h1. Por lo tanto, escribiríamos:



¿Qué es html?

HTML, siglas en inglés de HyperText Markup Language ('lenguaje de marcas de hipertexto'), hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros. Es un estándar a cargo del World Wide Web Consortium (W3C) o Consorcio WWW, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. Se considera el lenguaje web más importante siendo su invención crucial en la aparición, desarrollo y expansión de la World Wide Web (WWW). Es el estándar que se ha impuesto en la visualización de páginas web y es el que todos los navegadores actuales han adoptado.

El lenguaje HTML basa su filosofía de desarrollo en la diferenciación. Para añadir un elemento externo a la página (imagen, vídeo, script, entre otros.), este no se incrusta

directamente en el código de la página, sino que se hace una referencia a la ubicación de dicho elemento mediante texto. De este modo, la página web contiene solamente texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma (estándar) por cualquier navegador web actualizado.



¿Qué es LaTeX?

LaTeX es un sistema de composición de textos, orientado a la creación de documentos escritos que presenten una alta calidad tipográfica. Por sus características y posibilidades, es usado de forma especialmente intensa en la generación de artículos y libros científicos que incluyen, entre otros elementos, expresiones matemáticas.

LaTeX está formado por un gran conjunto de macros de TeX, escrito por Leslie Lamport en 1984, con la intención de facilitar el uso del lenguaje de composición tipográfica, Tex, creado por Donald Knuth. Es muy utilizado para la composición de artículos académicos, tesis y libros técnicos, dado que la calidad tipográfica de los documentos realizados en LaTeX, se considera adecuada a las necesidades de una editorial científica de primera línea, muchas de las cuales ya lo emplean.

LaTeX es software libre bajo licencia LPPL.

L^AT_EX

¿Que es csv?

Los archivos **CSV** (del inglés *comma-separated values*) son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas (o punto y coma en donde la coma es el separador decimal: Chile, Perú, Argentina, España, Brasil...) y las filas por saltos de línea.

El formato CSV es muy sencillo y no indica un juego de caracteres concreto, ni cómo van situados los bytes, ni el formato para el salto de línea. Estos puntos deben indicarse muchas veces al abrir el archivo, por ejemplo, con una hoja de cálculo.

El formato CSV no está estandarizado. La idea básica de separar los campos con una coma es muy clara, pero se vuelve complicada cuando el valor del campo también contienen comillas dobles o saltos de línea. Las implementaciones de CSV pueden no manejar esos datos, o usar comillas de otra clase para envolver el campo. Pero esto no resuelve el problema: algunos campos también necesitan embeber estas comillas, así que las implementaciones de CSV pueden incluir caracteres o secuencias de escape.

Además, el término "CSV" también denota otros formatos de valores separados por delimitadores que usan delimitadores diferentes a la coma (como los valores separados por tabuladores). Un delimitador que no está presente en los valores de los campos (como un tabulador) mantiene el formato simple. Estos archivos separados por delimitadores alternativos reciben en algunas ocasiones la extensión aunque este uso sea incorrecto. Esto puede causar problemas en el intercambio de datos, por ello muchas aplicaciones que usan archivos CSV tienen opciones para cambiar el carácter delimitador.



4. Si ha memorizado estos pasos y se siente con confianza, ejecute los pasos 1 a 3 para salir y volver a entrar al editor. Después mueva el cursor hasta la lección 1.3.

~~~~~

Lección 1.3: EDICIÓN DE TEXTO - BORRADO

"" Estando en modo Normal pulse `x` para borrar el carácter sobre el cursor. ""

1. Mueva el cursor a la línea de abajo señalada con `---`.
2. Para corregir los errores, mueva el cursor hasta que esté bajo el carácter que va a ser borrado.
3. Pulse la tecla `x` para borrar el carácter sobrante.
4. Repita los pasos 2 a 4 hasta que la frase sea la correcta.

---> la vaca saltó sobre la luna.

5. Ahora que la línea está correcta, continúe con la lección 1.4.

NOTA: A medida que vaya avanzando en este tutor no intente memorizar, aprenda practicando.

~~~~~

Lección 1.4: EDICIÓN DE TEXTO - INSERCIÓN

"" Estando en modo Normal pulse `i` para insertar texto. ""

1. Mueva el cursor a la primera línea de abajo señalada con `---`.
2. Para que la primera línea se igual a la segunda mueva el cursor bajo el primer carácter que sigue al texto que ha de ser insertado.

"" Estando en modo Normal pulse `i` para insertar texto. ""

1. Mueva el cursor a la primera línea de abajo señalada con `---`.
2. Para que la primera línea se igual a la segunda mueva el cursor bajo el primer carácter que sigue al texto que ha de ser insertado.
3. Pulse `i` y escriba los caracteres a añadir.
4. A medida que sea corregido cada error pulse `<ESC>` para volver al modo Normal. Repita los pasos 2 a 4 para corregir la frase.

---> Fita texto en esta .

---> Falta algo de texto en esta línea.

5. Cuando se sienta cómodo insertando texto pase al resumen que está más abajo.

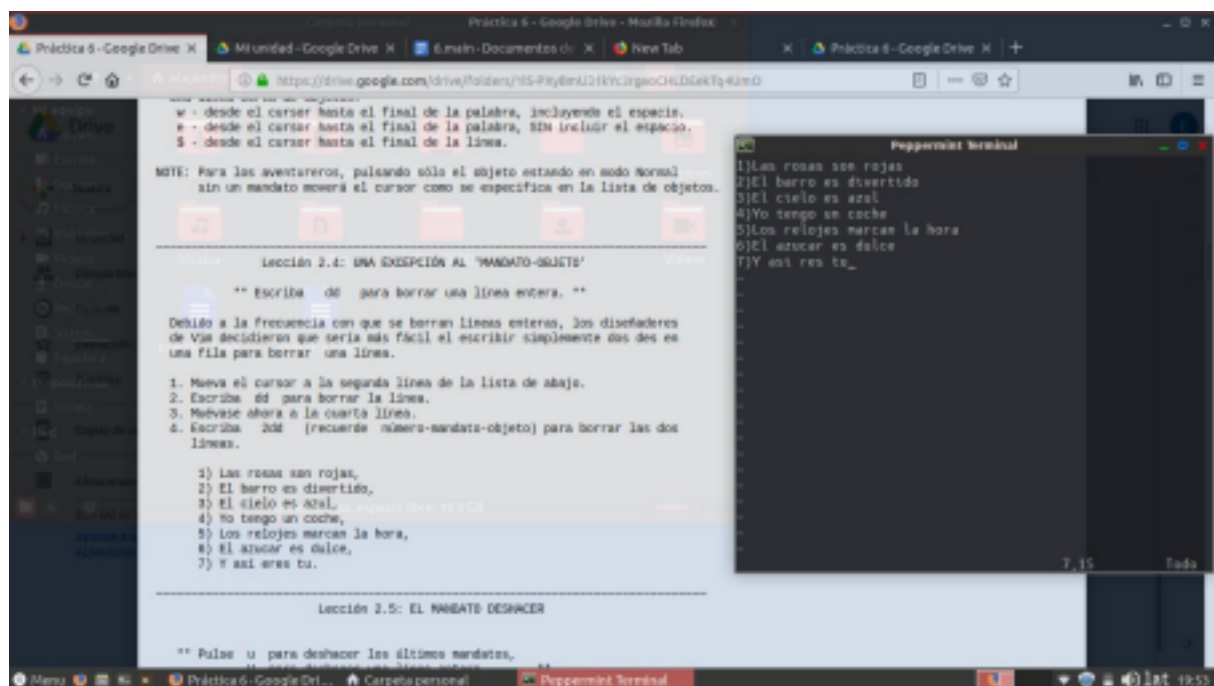
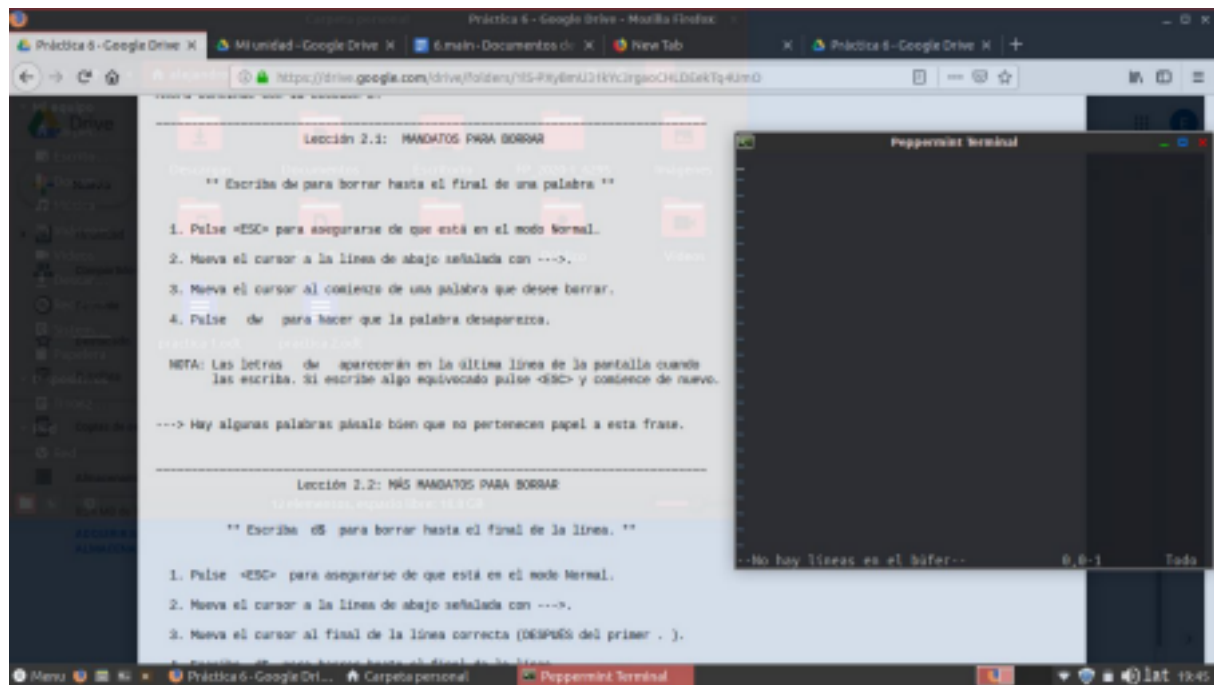
~~~~~

RESUMEN DE LA LECCIÓN 1

1. El cursor se mueve utilizando las teclas de las flechas o las teclas `h` (izquierda) `j` (abajo) `k` (arriba) `l` (derecha)
2. Para acceder a Vim (desde el símbolo del sistema `%`) escriba:  
`vim FILENAME <INTRO>`
3. Para salir de Vim escriba: `<ESC>` ;q; `<INTRO>` para eliminar todos los cambios.
4. Para borrar un carácter sobre el cursor en modo Normal pulse: `x`
5. Para insertar texto en la posición del cursor estando en modo Normal:  
pulse `i` escriba el texto pulse `<ESC>`









Práctica 6 - Google Drive - Mozilla Firefox

Práctica 6 - Google Drive X Mi Unidad - Google Drive X 6 main - Documentos de X New Tab X Práctica 6 - Google Drive X

https://drive.google.com/drive/folders/1S-Py6mU3IKVCJrgaoCH6D6aTq4Um0

### Lección 2.4: UNA EXCEPCIÓN AL "MANDATO-OBJETO"

**\*\* Escriba `dd` para borrar una línea entera. \*\***

Debido a la frecuencia con que se borran líneas enteras, los diseñadores de Vim decidieron que sería más fácil el escribir simplemente dos `des` en una fila para borrar una línea.

1. Mueva el cursor a la segunda línea de la lista de abajo.
2. Escriba `dd` para borrar la línea.
3. Muevase ahora a la cuarta línea.
4. Escriba `2dd` (recuerde número-mandato-objeto) para borrar las dos líneas.

**Lección 2.4: una excepción al "mandato-objeto"**

- 1) Las rosas son rojas,
- 2) El barro es divertido,
- 3) El cielo es azul,
- 4) No tengo un coche,
- 5) Los relojes marcan la hora,
- 6) El azúcar es dulce,
- 7) Y así eres tú.

### Lección 3.5: EL MANDATO DESHACER

**\*\* Pulse `u` para deshacer los últimos mandatos, `U` para deshacer una línea entera. \*\***

1. Mueva el cursor a la línea de abajo señalada con `---` y sitúelo bajo el primer error.
2. Pulse `x` para borrar el primer carácter erróneo.
3. Pulse ahora `u` para deshacer el último mandato ejecutado.
4. Ahora corrija todos los errores de la línea usando el mandato `x`.
5. Pulse ahora `U` mayúscula para devolver la línea a su estado original.
6. Pulse ahora `u` unas pocas veces para deshacer lo hecho por `U` y los mandatos previos.
7. Ahora pulse `CTRL-R` (mantenga pulsada la tecla `CTRL` y pulse `R`) unas pocas veces para volver a ejecutar los mandatos (deshacer lo deshacha).

`---` Corrija los errores de esta línea y vuelva a ponerlos con deshacer.

8. Estos mandatos son muy útiles. Ahora pase al resumen de la Lección 2.

### RESUMEN DE LA LECCIÓN 2

1. Para borrar desde el cursor hasta el final de una palabra pulse: `dw`
2. Para borrar desde el cursor hasta el final de una línea pulse: `dd`
3. Para borrar una línea entera pulse: `dd`
4. El formato de un mandato es como Normal es:  
[número] mandato objeto # mandato [número] objeto  
donde:  
número - es cuantas veces se ha de ejecutar el mandato  
mandato - es lo que hay que hacer, por ejemplo, `d` para borrar  
objeto - es sobre lo que el mandato va a operar, por ejemplo  
`w` (palabra), `$` (hasta el final de la línea), etc.
5. Para deshacer acciones previas pulse: `u` (u minúscula)  
Para deshacer todos los cambios de una línea pulse: `U` (U mayúscula)  
Para deshacer la deshecho pulse: `CTRL-R`

Peppermint Terminal

```
1) Las rosas son rojas
2) El barro es divertido
3) El cielo es azul
4) Los relojes marcan la hora
5) El azúcar es dulce
7) Y así eres tú.
```

Práctica 6 - Google Drive - Mozilla Firefox

Práctica 6 - Google Drive X Mi Unidad - Google Drive X 6 main - Documentos de X New Tab X Práctica 6 - Google Drive X

https://drive.google.com/drive/folders/1S-Py6mU3IKVCJrgaoCH6D6aTq4Um0

1. Mueva el cursor a la línea de abajo señalada con `---` y sitúelo bajo el primer error.
2. Pulse `x` para borrar el primer carácter erróneo.
3. Pulse ahora `u` para deshacer el último mandato ejecutado.
4. Ahora corrija todos los errores de la línea usando el mandato `x`.
5. Pulse ahora `U` mayúscula para devolver la línea a su estado original.
6. Pulse ahora `u` unas pocas veces para deshacer lo hecho por `U` y los mandatos previos.
7. Ahora pulse `CTRL-R` (mantenga pulsada la tecla `CTRL` y pulse `R`) unas pocas veces para volver a ejecutar los mandatos (deshacer lo deshacha).

`---` Corrija los errores de esta línea y vuelva a ponerlos con deshacer.

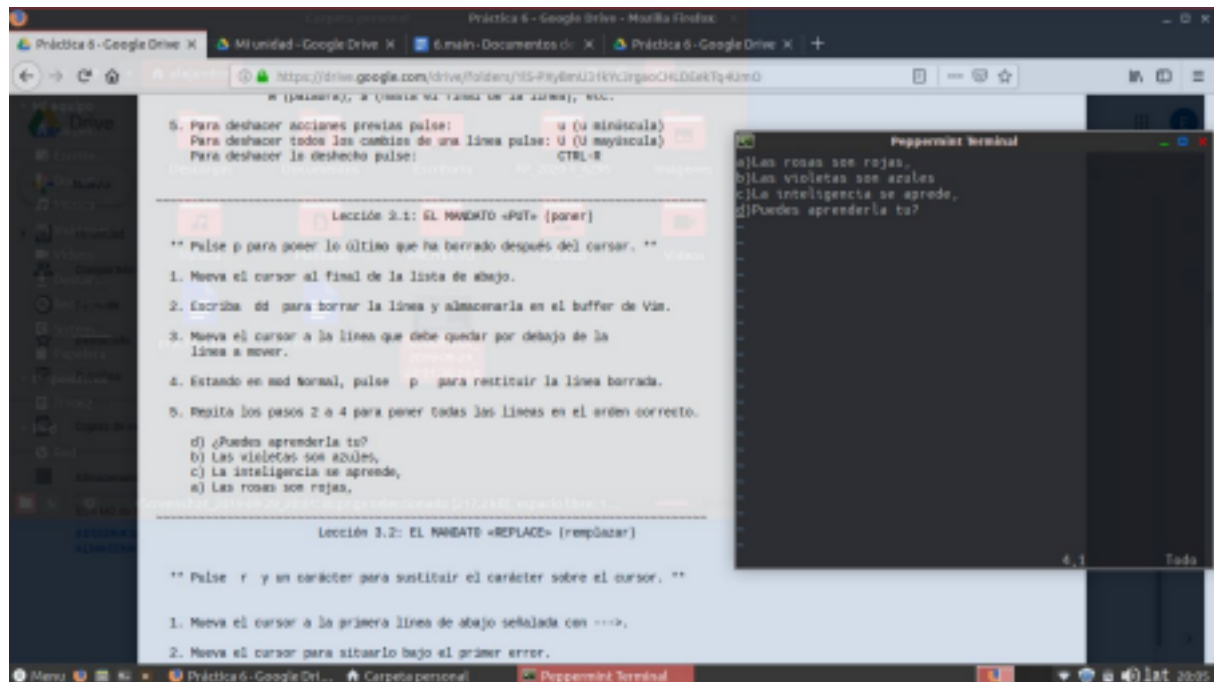
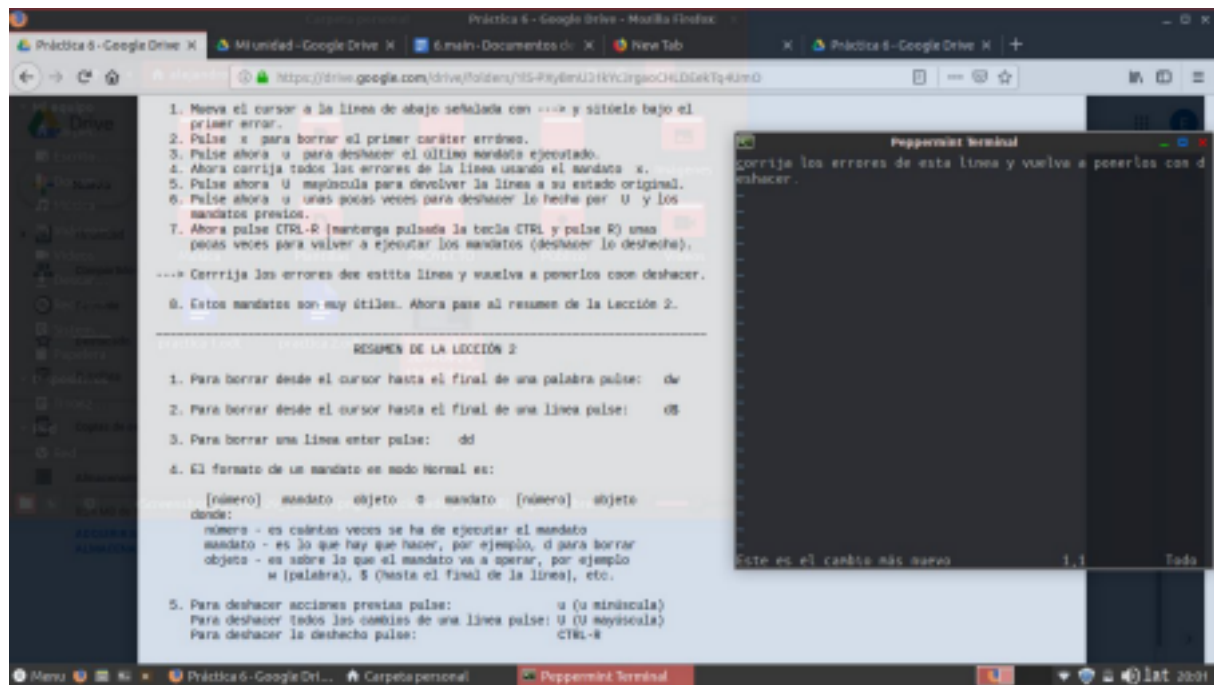
8. Estos mandatos son muy útiles. Ahora pase al resumen de la Lección 2.

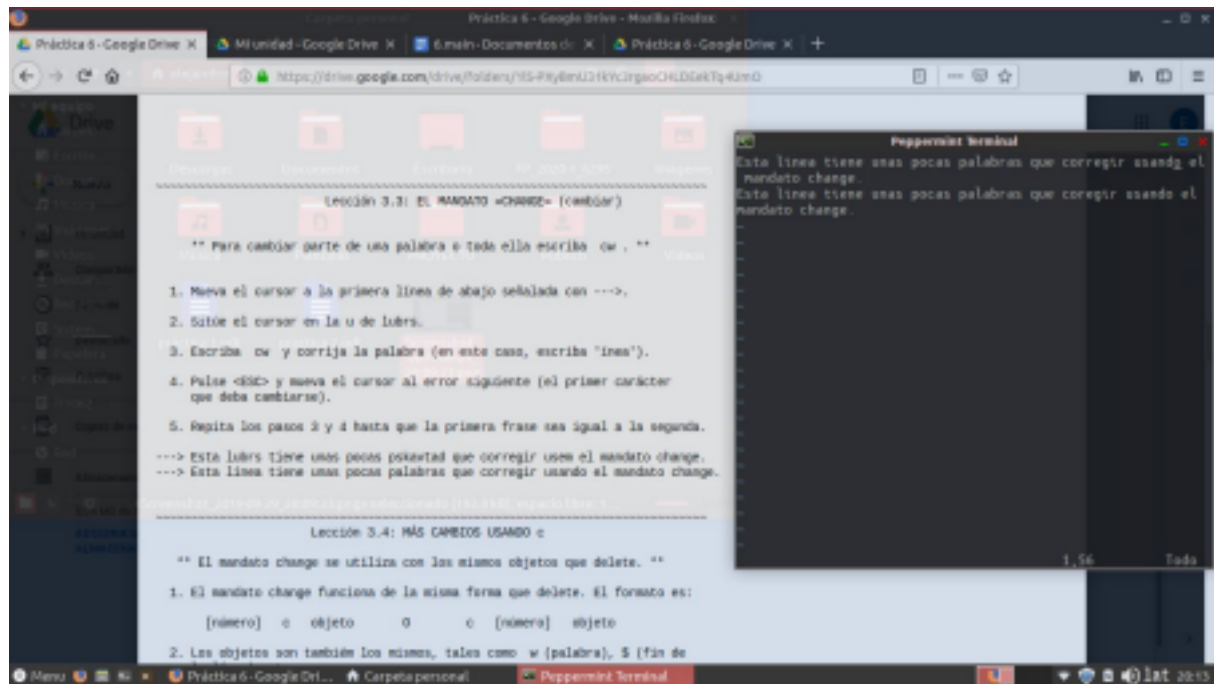
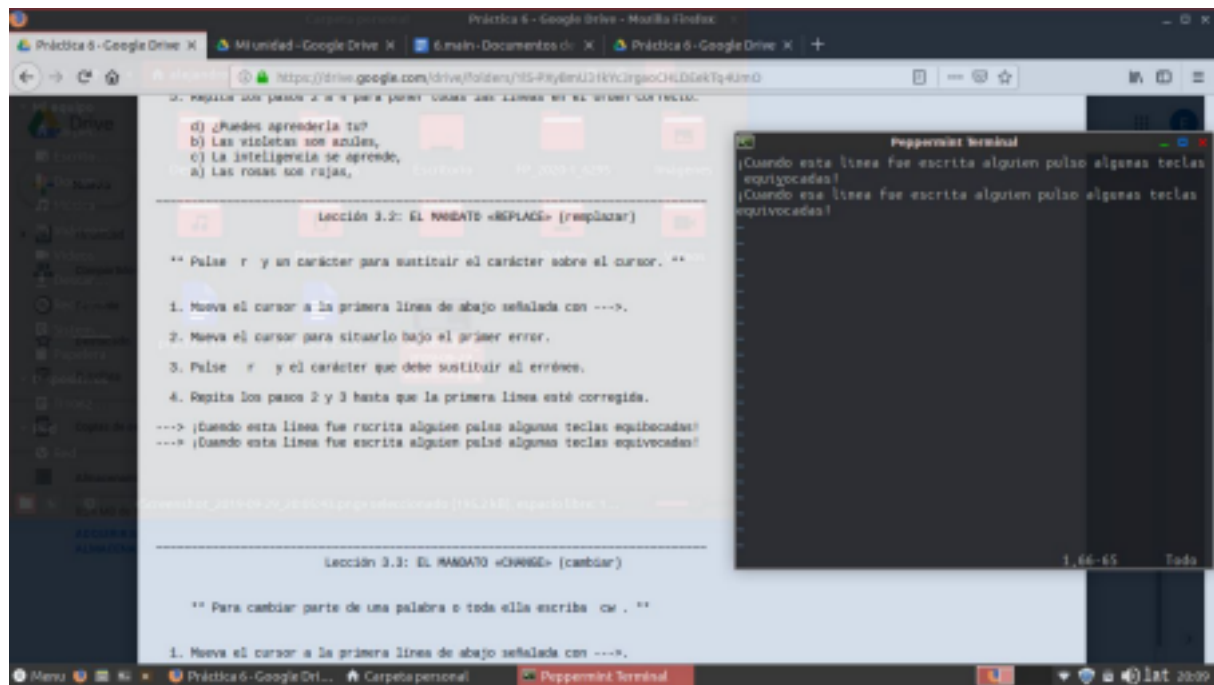
### RESUMEN DE LA LECCIÓN 2

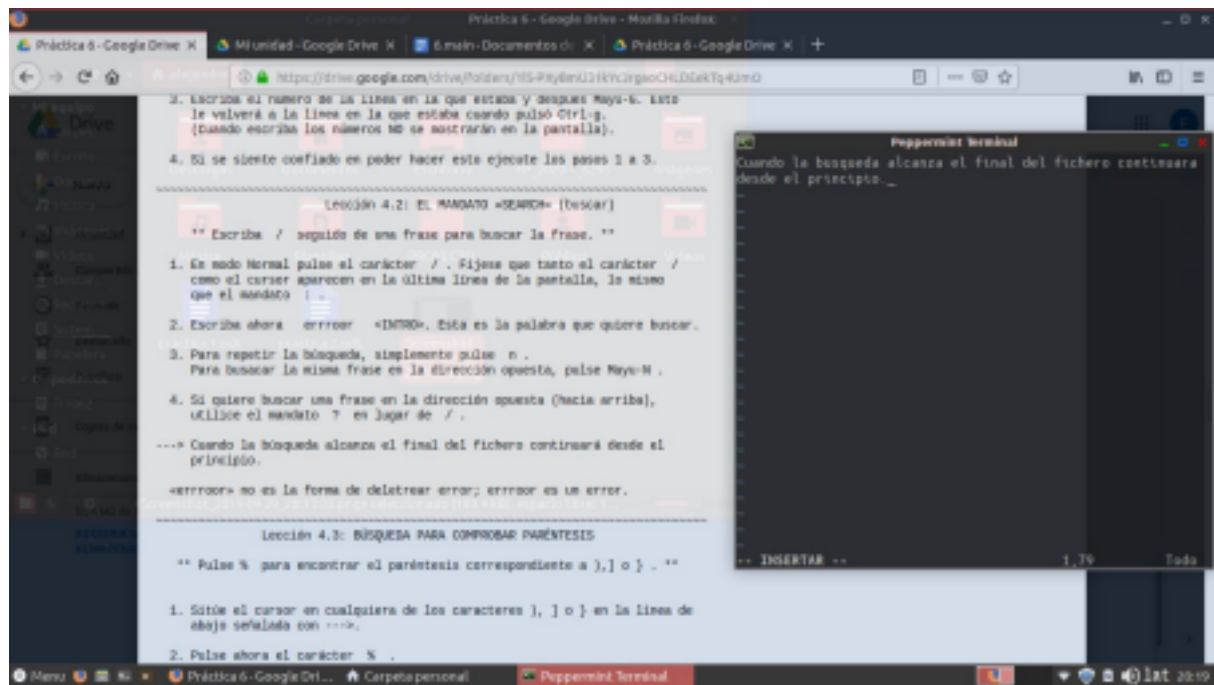
1. Para borrar desde el cursor hasta el final de una palabra pulse: `dw`
2. Para borrar desde el cursor hasta el final de una línea pulse: `dd`
3. Para borrar una línea entera pulse: `dd`
4. El formato de un mandato es como Normal es:  
[número] mandato objeto # mandato [número] objeto  
donde:  
número - es cuantas veces se ha de ejecutar el mandato  
mandato - es lo que hay que hacer, por ejemplo, `d` para borrar  
objeto - es sobre lo que el mandato va a operar, por ejemplo  
`w` (palabra), `$` (hasta el final de la línea), etc.
5. Para deshacer acciones previas pulse: `u` (u minúscula)  
Para deshacer todos los cambios de una línea pulse: `U` (U mayúscula)  
Para deshacer la deshecho pulse: `CTRL-R`

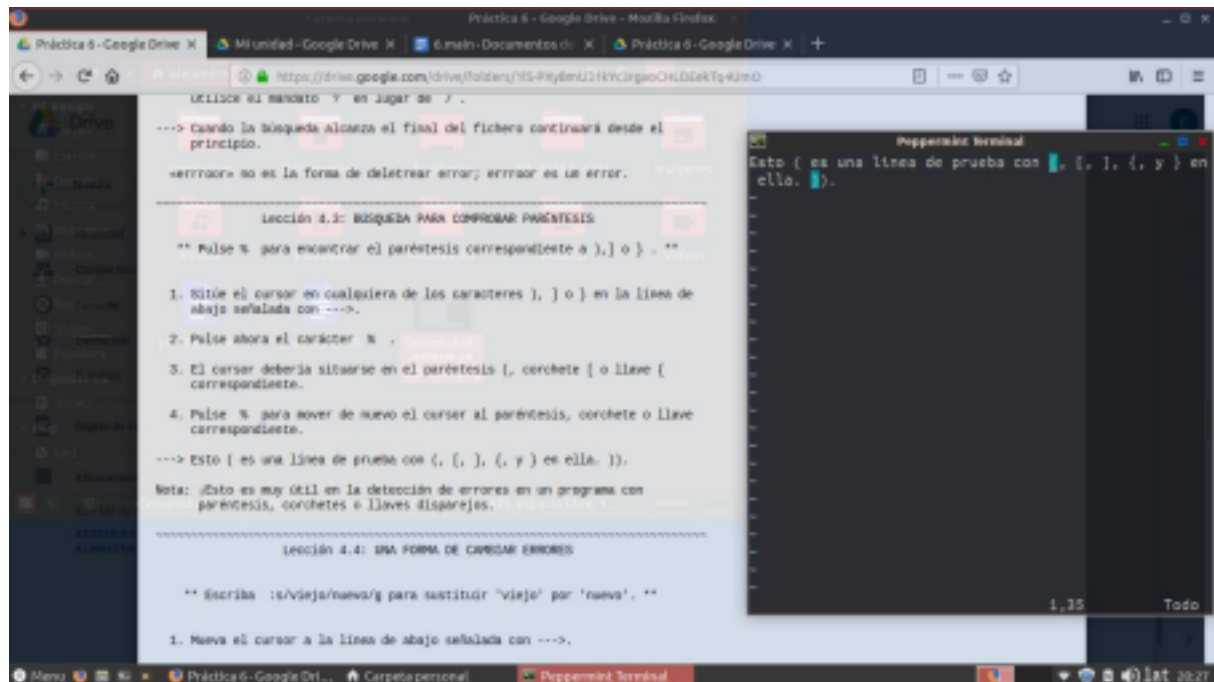
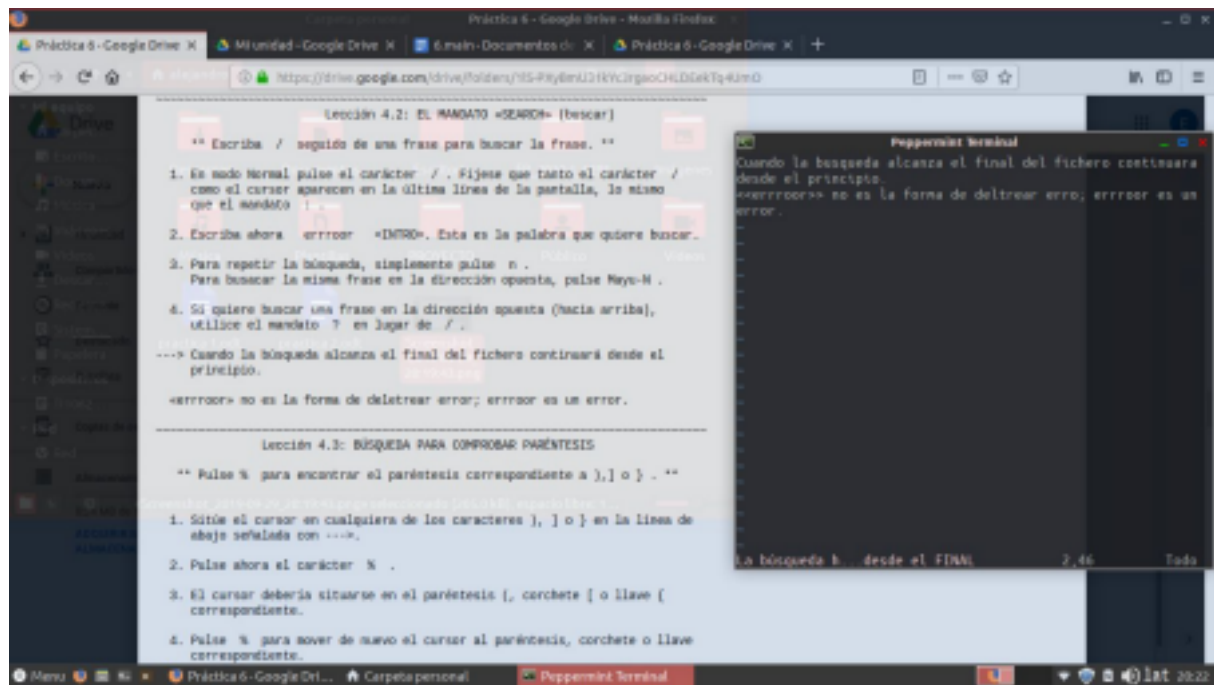
Peppermint Terminal

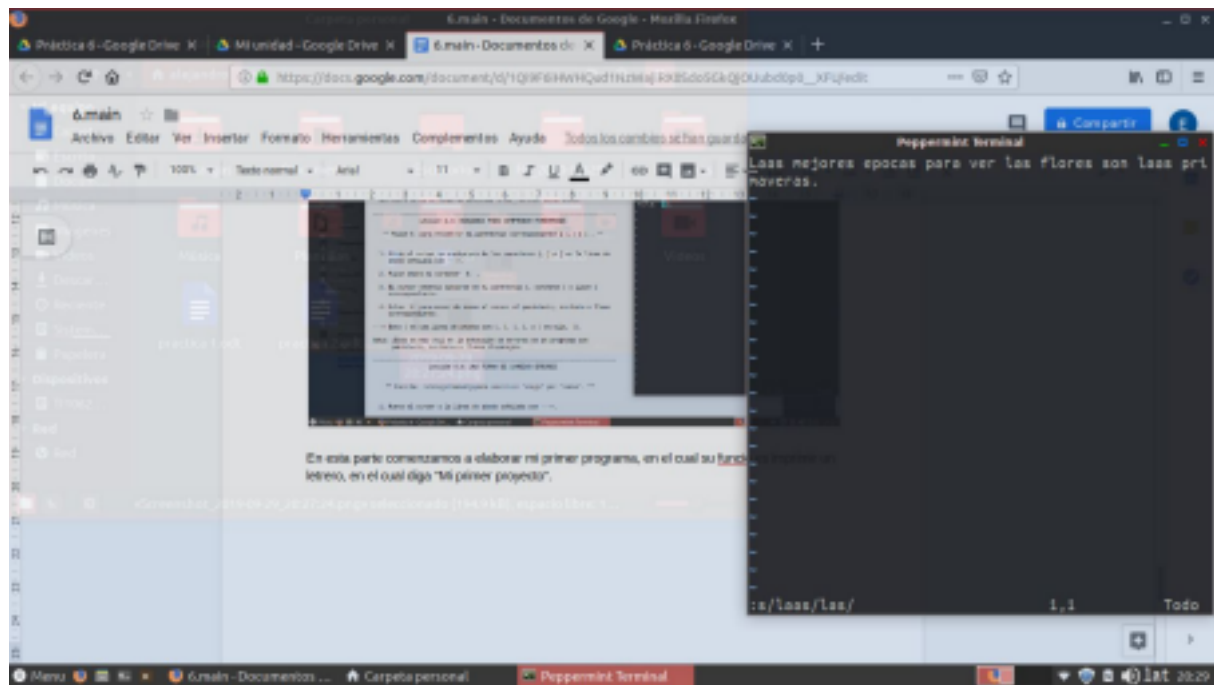
```
corrija los errores de esta línea y vuelva a ponerlos con deshacer.
```



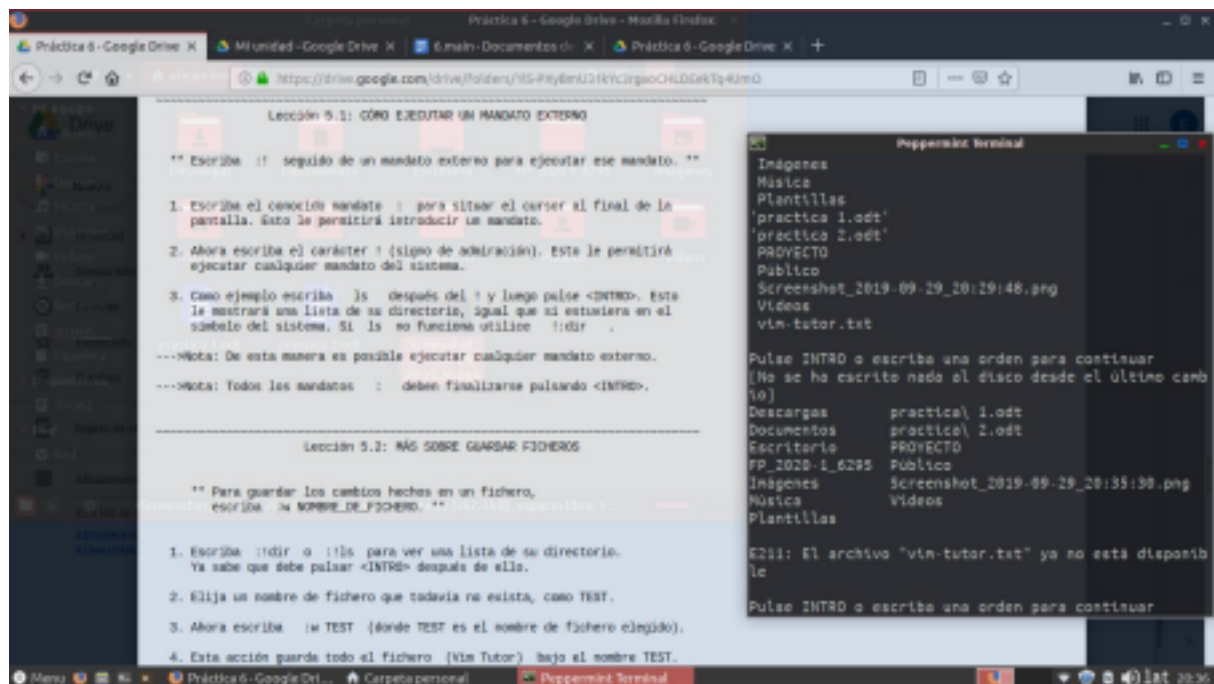
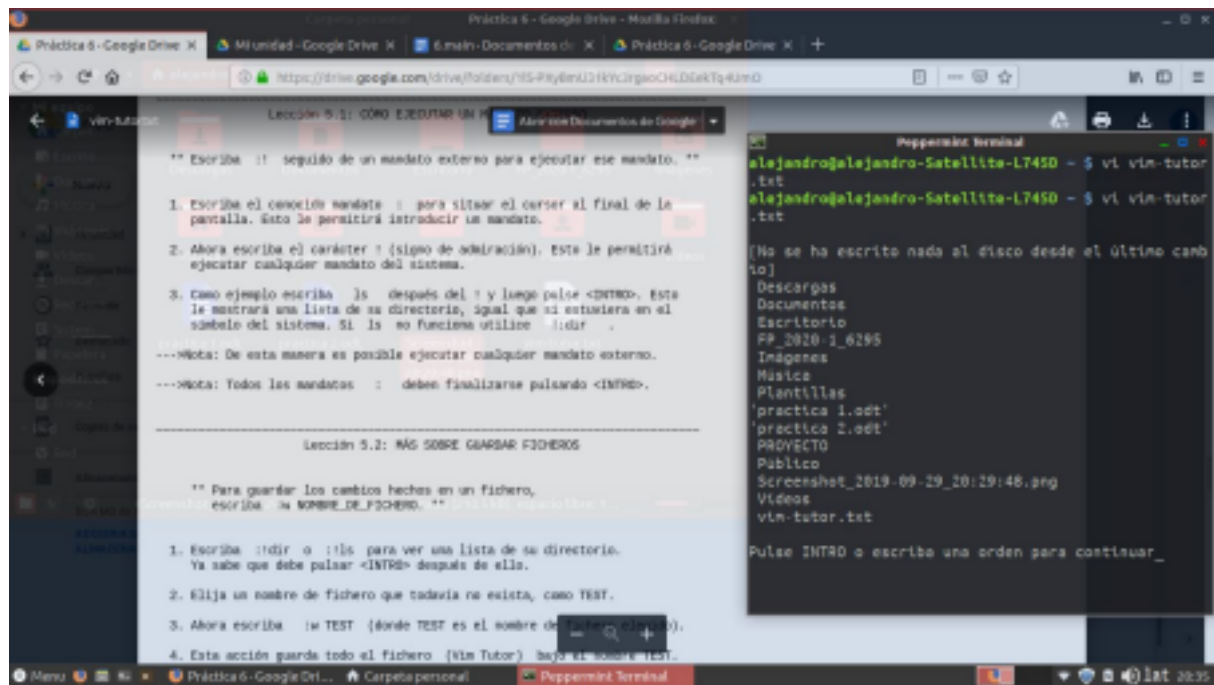












En esta parte comenzamos a elaborar mi primer programa, en el cual su función es imprimir un letrero, en el cual diga “Mi primer proyecto”.



```
fp03alu03 — -bash — 80x24
Namibia25:~ fp03alu03$ cd/
-bash: cd/: No such file or directory
Namibia25:~ fp03alu03$ ls
Desktop      Downloads    Movies       Pictures     int main()
Documents    Library      Music        Public       main.c
Namibia25:~ fp03alu03$ main.c
-bash: main.c: command not found
Namibia25:~ fp03alu03$ gcc main.c -o praxctica
main.c:3:2: warning: implicitly declaring library function 'printf' with type
      'int (const char *, ...)' [-Wimplicit-function-declaration]
      printf("Mi primer programa\n") ;
      ^
main.c:3:2: note: include the header <stdio.h> or explicitly provide a
      declaration for 'printf'
1 warning generated.
Namibia25:~ fp03alu03$ gcc main.c -o main
main.c:3:2: warning: implicitly declaring library function 'printf' with type
      'int (const char *, ...)' [-Wimplicit-function-declaration]
      printf("Mi primer programa\n") ;
      ^
main.c:3:2: note: include the header <stdio.h> or explicitly provide a
      declaration for 'printf'
1 warning generated.
Namibia25:~ fp03alu03$
```

```
main.c
UNREGISTERED
main.c
1 int main()
2 {
3     printf("Mi primer programa\n") ;
4
5     return 0 ;
6 }
```

Line 6, Column 2      Tab Size: 4      C

```
fp03alu03 — -bash — 80x24
[Namibia25:~ fp03alu03$ ls
Desktop      Downloads    Movies       Pictures     int main()
Documents    Library      Music        Public       main.c
[Namibia25:~ fp03alu03$ main.c
-bash: main.c: command not found
[Namibia25:~ fp03alu03$ gcc main.c -o praxctica
main.c:3:2: warning: implicitly declaring library function 'printf' with type
      'int (const char *, ...)' [-Wimplicit-function-declaration]
      printf("Mi primer programa\n") ;
      ^
main.c:3:2: note: include the header <stdio.h> or explicitly provide a
      declaration for 'printf'
1 warning generated.
[Namibia25:~ fp03alu03$ gcc main.c -o main
main.c:3:2: warning: implicitly declaring library function 'printf' with type
      'int (const char *, ...)' [-Wimplicit-function-declaration]
      printf("Mi primer programa\n") ;
      ^
main.c:3:2: note: include the header <stdio.h> or explicitly provide a
      declaration for 'printf'
1 warning generated.
[Namibia25:~ fp03alu03$ ./main
Mi primer programa
[Namibia25:~ fp03alu03$ ]
```

## CONCLUSIÓN

En esta práctica fue muy interesante, ya que vi las funciones que puede realizar "vi", además hice mi primer programa fue muy padre esta práctica.