



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* ALEJANDRO ESTEBAN PIMENTEL ALARCON

*Asignatura:* FUNDAMENTOS DE PROGRAMACIÓN

*Grupo:* 3

*No de Práctica(s):* 6

*Integrante(s):* MIRANDA GUTIERREZ CELINE

*No. de Equipo de  
cómputo empleado:* 41

*No. de Lista o Brigada:* 30

*Semestre:* 2020-1

*Fecha de entrega:* 30/09/2019

*Observaciones:* Para la actividad de vim tutor tenías que descargar el archivo que estaba en la carpeta de drive, abrirlo con vi y seguir las instrucciones de dicho archivo. ~~Debes asegurarte de que los~~ archivos estén en la carpeta de trabajo de la terminal.

CALIFICACIÓN: 9

**PRÁCTICA 6**  
**“ENTORNO DE C”**

**INTRODUCCIÓN**

El lenguaje de programación C fue creado por Brian Kernighan y Dennis Ritchie a mediados de los años 70. El lenguaje C es un lenguaje estructurado, en el mismo sentido que lo son otros lenguajes de programación tales como el lenguaje Pascal, el Ada o el Modula-2, pero no es estructurado por bloques, o sea, no es posible declarar subrutinas (pequeños trozos de programa) dentro de otras subrutinas, a diferencia de como sucede con otros lenguajes estructurados tales como el Pascal. Además, el lenguaje C no es rígido en la comprobación de tipos de datos, permitiendo fácilmente la conversión entre diferentes tipos de datos y la asignación entre tipos de datos diferentes, por ejemplo la expresión siguiente es válida en C: float a; /\*Declaro una variable para numeros reales\*/ int b; /\*Declaro otra variable para numero enteros\*/ b=a; /\*Asigno a la variable para entera el numero real\*/ Todo programa de C consta, básicamente, de un conjunto de funciones, y una función llamada main, la cual es la primera que se ejecuta al comenzar el programa, llamándose desde ella al resto de funciones que compongan nuestro programa.

El lenguaje C posee un número reducido de palabras reservadas (tan solo 32) que define el standard ANSI-C. Estas palabras reservadas pueden verse en la tabla siguiente:

signed	sizeof	static	struct	switch	typedef	union
unsigned	void	volatile	while			

*Tabla 1.1: Palabras reservadas del lenguaje C.*

auto	break	case	char	const	continue	default
do	double	else	enum	extern	float	for
goto	if	int	long	register	return	short

**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.

**ACTIVIDAD 1**

El texto plano es el tipo de archivo más sencillo que hay, ya que en su contenido no hay otra cosa más que el texto que lo conforma. Hacer una investigación muy somera acerca de los siguientes tipos de archivos:

- **txt**

La extensión TXT representa "textfile" (archivo de texto), que sustituyó a su antiguo nombre "flatfile" (archivo sin formato). Este archivo informático estructura series de líneas de texto. El final del archivo se identifica habitualmente con un carácter especial definido como un marcador "end-of-file" (final de archivo), ubicado a continuación de la última línea de texto. Este tipo de contenedor incluye texto sin formato, pero tiene otras capacidades. Los archivos TXT sirven como almacenes de información a la vez que evitan las complicaciones propias de otros formatos de archivo. Los archivos afectados por la corrupción de datos son fácilmente recuperables y el usuario puede continuar su trabajo con la información restante. El inconveniente de utilizar archivos TXT proviene de su baja entropía, que provoca que los archivos TXT ocupen más espacio que otros archivos de texto.

Los archivos TXT contienen un formato mínimo, pero cumplen con las definiciones de formato aceptadas por la terminal del sistema y los editores de texto simple. Los archivos TXT son universales debido a que cualquier procesador de texto puede leerlos. Los archivos TXT pueden utilizar Unicode para facilitar su lectura a usuarios de distintos idiomas. Los archivos de texto exclusivamente en código ASCII pueden intercambiarse y leerse en Unix, Mac y Windows. El conjunto de caracteres más habitual es UTF-8, se diferencia del código ASCII en la marca de orden de bytes y es retrocompatible con ASCII.

- **Markdown**

Markdown fue desarrollado en 2004 por John Gruber, y se refiere tanto a (1) una manera de formar archivos de texto, como a (2) una utilidad del lenguaje de programación Perl para convertir archivos Markdown en HTML. En esta lección nos centraremos en la primera acepción y aprenderemos a escribir archivos utilizando la sintaxis de Markdown.

Los archivos de texto plano tienen muchas ventajas sobre otro tipo de formato. Por un lado, se pueden leer prácticamente en todos los dispositivos. También han resistido la prueba del paso del tiempo mejor que otro tipo de archivos -si alguna vez has intentado abrir un documento guardado en un formato de procesador de textos heredado, estarás familiarizado con los problemas de compatibilidad que implican-

Al utilizar la sintaxis de Markdown, serás capaz de producir archivos que pueden ser legibles como texto plano y que a la vez están listos para ser formados en otras plataformas. Muchos generadores de bitácoras y de sitios estáticos, así como sitios como GitHub, también aceptan Markdown y traducen estos archivos a HTML para su visualización en la web. Además, herramientas como Pandoc pueden convertir archivos en o desde Markdown. Para más información sobre Pandoc puedes consultar la lección sobre Escritura sostenible utilizando Pandoc y Markdown de Dennise Tenen y Grant Wythoff.

- **html**

HTML significa "Lenguaje de Marcado de Hypertexto" por sus siglas en inglés "HyperText Markup Language", es un lenguaje que pertenece a la familia de los "lenguajes de marcado" y es utilizado para la elaboración de páginas web. El estándar HTML lo define la W3C (World Wide Web Consortium) y actualmente HTML se encuentra en su versión HTML5.

Cabe destacar que HTML no es un lenguaje de programación ya que no cuenta con funciones aritméticas, variables o estructuras de control propias de los lenguajes de programación, por lo que HTML genera únicamente páginas web estáticas, sin embargo, HTML se puede usar en conjunto con diversos lenguajes de programación para la creación de páginas web dinámicas.

El lenguaje HTML sirve para describir la estructura básica de una página y organizar la forma en que se mostrará su contenido, además de que HTML permite incluir enlaces (links) hacia otras páginas o documentos,

HTML es un lenguaje de marcado descriptivo que se escribe en forma de etiquetas para definir la estructura de una página web y su contenido como texto, imágenes, entre otros, de modo que HTML es el encargado de describir (hasta cierto punto) la apariencia que tendrá la página web.

- **LaTeX**

Es un sistema de composición de textos que está orientado especialmente a la creación de documentos científicos que contengan formulas matemáticas, cuadros y tablas. Además, también se pueden crear otros tipos de documentos, que pueden ser desde cartas sencillas hasta libros completos. LATEX está organizado sobre TEX. LATEX está disponible para la mayoría de los miniordenadores y microordenadores, desde IBM PCs en adelante.

LaTeX es un sistema de composición de textos que está formado mayoritariamente por órdenes (macros) construidas a partir de comandos de TeX un lenguaje «de bajo nivel», en el sentido de que sus acciones últimas son muy elementales pero con la ventaja añadida, en palabras de Lamport de «poder aumentar las capacidades de LaTeX utilizando comandos propios del TeX descritos en The TeXbook». Esto es lo que convierte a LaTeX en una herramienta práctica y útil pues, a su facilidad de uso, se une toda la potencia de TeX. Estas características hicieron que LaTeX se extendiese rápidamente entre un amplio sector científico y técnico, hasta el punto de convertirse en uso obligado en comunicaciones y congresos, y requerido por determinadas revistas a la hora de entregar artículos académicos. Su código abierto permitió que muchos usuarios realizasen nuevas utilidades que extendiesen sus capacidades con objetivos muy variados, a veces ajenos a la intención con la que fue creado LaTeX no es un programa donde se pueda escribir directamente un documento. En LaTeX, primero se escribe el código fuente en cualquier editor de textos, después se compila, esto genera un archivo.dvi, que puede ser visualizado como resultado final, además se puede transformar a otros formatos como PostScript (.ps), pdf (.pdf) o HTML por ejemplo.

- **Csv**

Un archivo CSV es un archivo de texto que contiene algunos datos. En circunstancias normales, un archivo CSV se utiliza para transferir datos de una aplicación a otra. A modo de explicación, un archivo CSV almacena datos, tanto números como texto en un texto sin formato. Como sabrá, un texto sin formato comprime la información y permite el formato del texto. **La extensión de archivos .CSV permite incluir o agregar cualquier valor**, como por ejemplo números de

Normalmente, todos los campos están separados por comas, mientras que todos los registros están separados por una línea de caracteres elaborada. En las hojas de Excel, todos los campos que tienen una coma, están incrustados entre comillas dobles que se conocen comúnmente como calificadores de texto, es decir, una sola celda con verde, azul y amarillo, se capturarán como "verde, azul y amarillo".

## ACTIVIDAD 2

En esta actividad lo que hice fue probar cómo funcionaba la edición de textos con Vim, en la terminal puse **\$ vim vim-tutor.txt** el cual me abrió una pantalla en negro, con la tecla **I** se le da una instrucción a la terminal de que se insertará texto nuevo. Escribí mi texto procurando que fueras más de dos líneas. Cuando quise salir del modo edición fue con la tecla **ESC**. Después con la flechas del teclado subí al principio de mi texto y presioné la tecla **V**. En la parte inferior me apareció algo nuevo y decía ahora “VISUAL”. Jugué un poco con esto seleccionando dos líneas, recortando y volviendo a pegar para ver el funcionamiento de las letras “P”, “Y” y “X”.

[illegible]

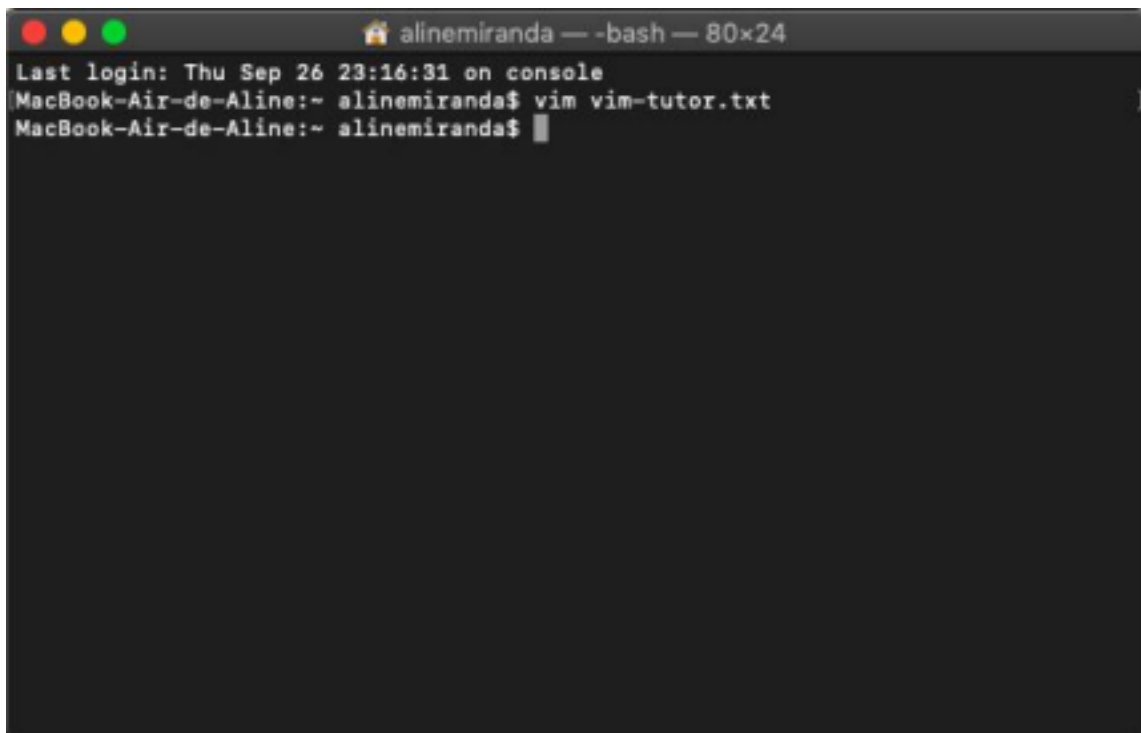


```
alinemiranda — vim vim-tutor.txt — 80x24
Esto es una prueba para aprender a utilizar vim e intentar editar textos.
Espero que si me ponga mi diez porque me costó entenderle
Saludos cordiales.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
3 lines yanked
```

```
alinemiranda — vim vim-tutor.txt — 80x24
Esto es una prueba para aprender a utilizar vim e intentar editar textos.
Espero que si me ponga mi diez porque me costó entenderle
Esto es una prueba para aprender a utilizar vim e intentar editar textos.
Espero que si me ponga mi diez porque me costó entenderle
Saludos cordiales.
Saludos cordiales.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
3 more lines
```

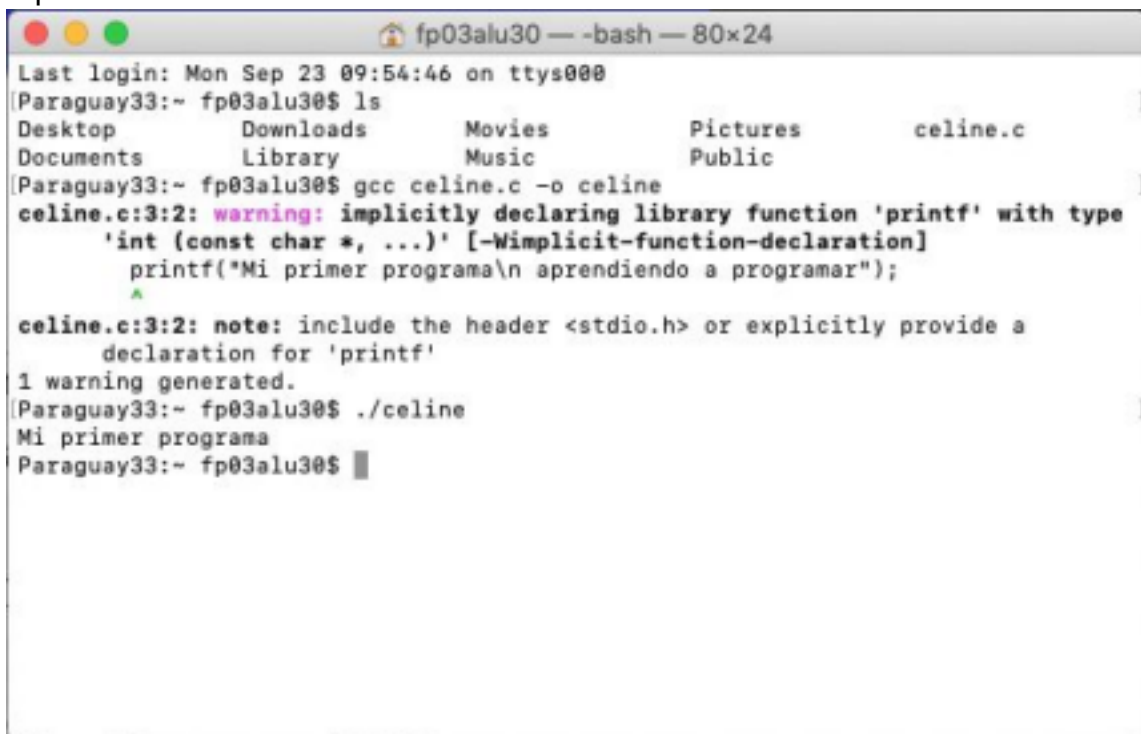




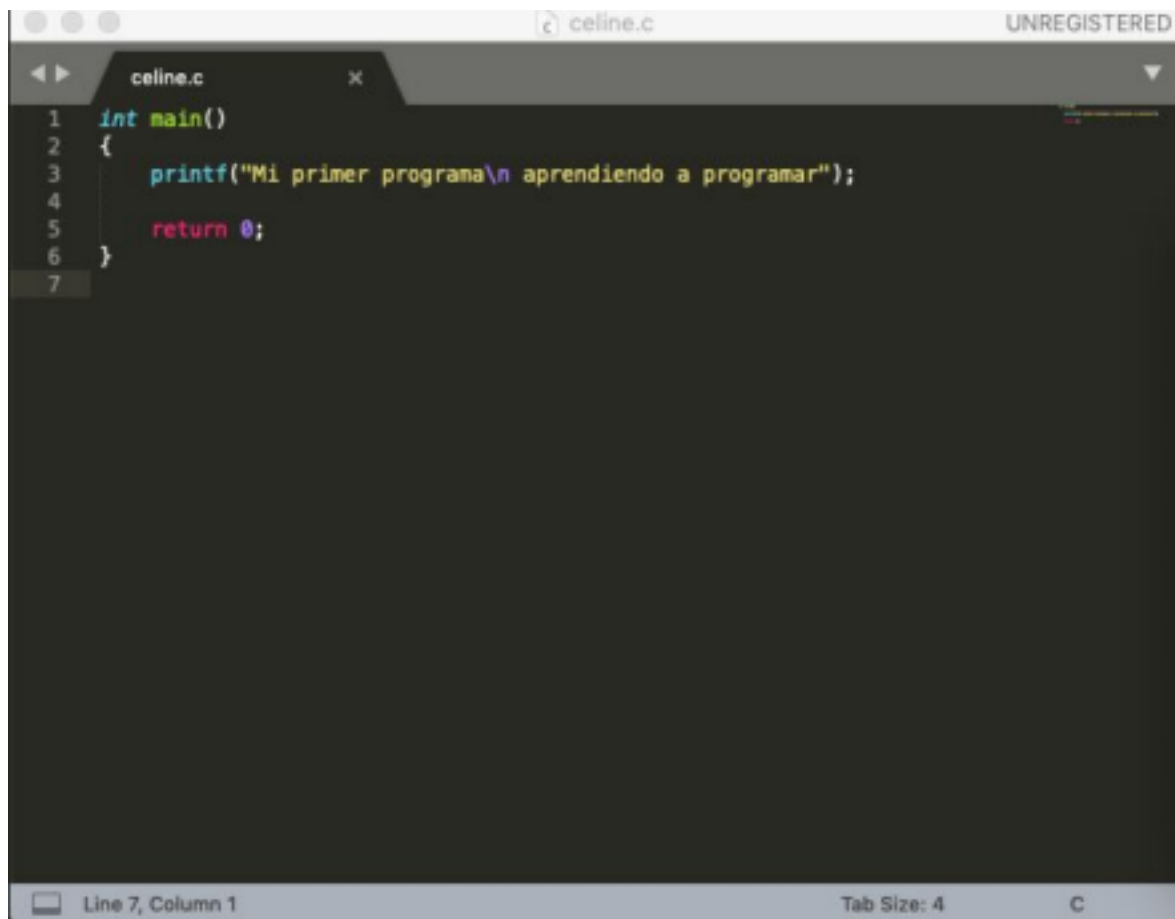


```
alinemiranda — -bash — 80x24
Last login: Thu Sep 26 23:16:31 on console
MacBook-Air-de-Aline:~ alinemiranda$ vim vim-tutor.txt
MacBook-Air-de-Aline:~ alinemiranda$
```

Después se trabajó en Notepad++ que es un editor gráfico mucho más parecido a lo que estamos acostumbrados a trabajar en clase y con lo cual va más encaminado a la creación de nuevos programas que resuelvan problemas en específico.



```
fp03alu30 — -bash — 80x24
Last login: Mon Sep 23 09:54:46 on ttys000
Paraguay33:~ fp03alu30$ ls
Desktop      Downloads    Movies       Pictures     celine.c
Documents    Library      Music        Public
Paraguay33:~ fp03alu30$ gcc celine.c -o celine
celine.c:3:2: warning: implicitly declaring library function 'printf' with type
      'int (const char *, ...)' [-Wimplicit-function-declaration]
      printf("Mi primer programa\n aprendiendo a programar");
      ^
celine.c:3:2: note: include the header <stdio.h> or explicitly provide a
      declaration for 'printf'
1 warning generated.
Paraguay33:~ fp03alu30$ ./celine
Mi primer programa
Paraguay33:~ fp03alu30$
```



```
celine.c UNREGISTERED
1 int main()
2 {
3     printf("Mi primer programa\n aprendiendo a programar");
4
5     return 0;
6 }
7
```

Line 7, Column 1 Tab Size: 4 C

## CONCLUSIONES

El conocer más los tipos de archivos nos permite ampliar nuestro panorama y conocer qué tipo de archivo se adecua dependiendo a nuestros objetivos. Como todo, es cuestión de práctica e indagar un poco más para aplicar los nuevos conocimientos en otras materias y en un futuro incluso podría auxiliarnos con la tesis. Considero que es una práctica muy útil para extensión de conocimientos sobre programas.