



INSTITUTO DE FORMACIÓN TÉCNICA SUPERIOR N°18

<http://www.ifts18.edu.ar>

Mansilla 3643 (C1425BBW), Ciudad Autónoma de Buenos Aires

TÉCNICO SUPERIOR EN ANÁLISIS DE SISTEMAS

Examen Final

Asignatura:

DIAGRAMACIÓN LÓGICA

Autor:

PROF. LEANDRO E. COLOMBO VIÑA

última modificación:

5 de julio de 2018

Índice general

1 Examen	1
1.1 Introducción	1
1.2 Funcionamiento	1
1.2.1 Página del manual	2
1.3 Consiga	3
1.4 Criterios de aprobación	4
1.4.1 Programa	4
1.4.2 Examen	4
2 Recursos y sugerencias	5
2.1 Lorem Ipsum	5

Resumen

El presente documento contiene las consignas y consideraciones para la realización del examen parcial de la asignatura Diagramación Lógica que se dicta durante el 1° cuatrimestre del primer año de la carrera Técnico Superior en Análisis de Sistemas en el Instituto de Formación Técnica Superior N°18. Esta consigna es válida para la cursada correspondiente al año 2018.

Capítulo 1

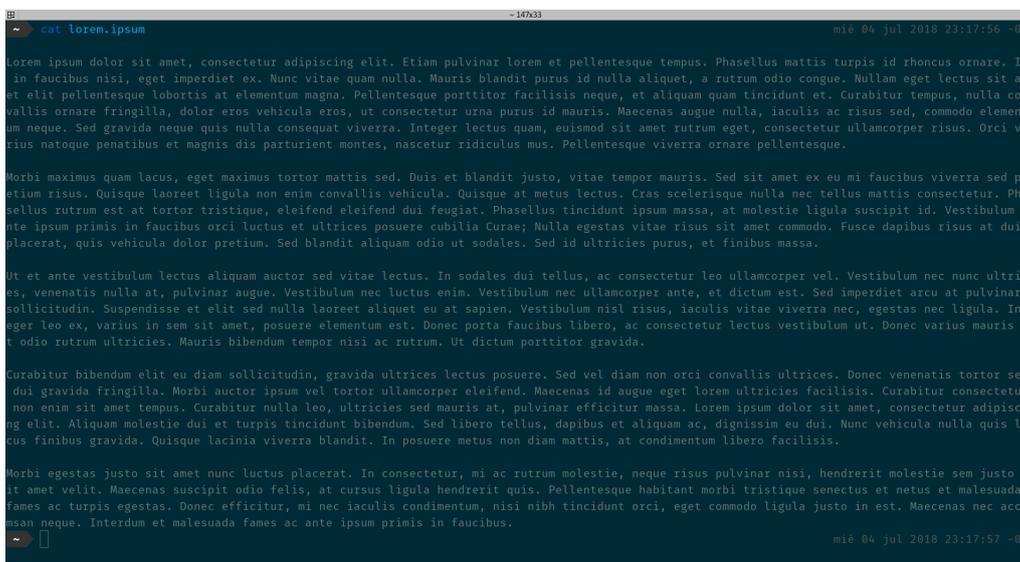
Examen

1.1 Introducción

En los sistemas operativos *nix generalmente vienen muchas utilidades de terminal que son para el manejo de archivos de texto. En el set de herramientas suele venir el comando `wc`. El nombre del comando es un acrónimo de *Word Count*, que significa contador de palabras en inglés.

1.2 Funcionamiento

Supongamos que tenemos un archivo de texto que se llama `lorem.ipsium`. Podemos visualizar su contenido con un editor de texto del entorno gráfico, o también con otra de las utilidades que viene en el sistema como `cat`.



```
mié 04 jul 2018 23:17:56 -63
~ cat lorem.ipsium
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam pulvinar lorem et pellentesque tempus. Phasellus mattis turpis id rhoncus ornare. In
in faucibus nisi, eget imperdiet ex. Nunc vitae quam nulla. Mauris blandit purus id nulla aliquet, a rutrum odio congue. Nullam eget lectus sit am
et elit pellentesque lobortis at elementum magna. Pellentesque porttitor facilisis neque, et aliquam quam tincidunt et. Curabitur tempus, nulla con
vallis ornare fringilla, dolor eros vehicula eros, ut consectetur urna purus id mauris. Maecenas augue nulla, iaculis ac risus sed, commodo element
um neque. Sed gravida neque quis nulla consequat viverra. Integer lectus quam, euismod sit amet rutrum eget, consectetur ullamcorper risus. Orci va
rius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Pellentesque viverra ornare pellentesque.

Morbi maximus quam lacus, eget maximus tortor mattis sed. Duis et blandit justo, vitae tempor mauris. Sed sit amet ex eu mi faucibus viverra sed pr
etium risus. Quisque laoreet ligula non enim convallis vehicula. Quisque at metus lectus. Cras scelerisque nulla nec tellus mattis consectetur. Pha
sellus rutrum est at tortor tristique, eleifend eleifend dui feugiat. Phasellus tincidunt ipsum massa, at molestie ligula suscipit id. Vestibulum a
nte ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nulla egestas vitae risus sit amet commodo. Fusce dapibus risus at dui
placerat, quis vehicula dolor pretium. Sed blandit aliquam odio ut sodales. Sed id ultricies purus, et finibus massa.

Ut et ante vestibulum lectus aliquam auctor sed vitae lectus. In sodales dui tellus, ac consectetur leo ullamcorper vel. Vestibulum nec nunc ultric
es, venenatis nulla at, pulvinar augue. Vestibulum nec luctus enim. Vestibulum nec ullamcorper ante, et dictum est. Sed imperdiet arcu at pulvinar
sollicitudin. Suspendisse et elit sed nulla laoreet aliquet eu at sapien. Vestibulum nisl risus, iaculis vitae viverra nec, egestas nec ligula. Int
eger leo ex, varius in sem sit amet, posuere elementum est. Donec porta faucibus libero, ac consectetur lectus vestibulum ut. Donec varius mauris a
t odio rutrum ultricies. Mauris bibendum tempor nisi ac rutrum. Ut dictum porttitor gravida.

Curabitur bibendum elit eu diam sollicitudin, gravida ultrices lectus posuere. Sed vel diam non orci convallis ultrices. Donec venenatis tortor sed
dui gravida fringilla. Morbi auctor ipsum vel tortor ullamcorper eleifend. Maecenas id augue eget lorem ultrices facilisis. Curabitur consectetur
non enim sit amet tempus. Curabitur nulla leo, ultricies sed mauris at, pulvinar efficitur massa. Lorem ipsum dolor sit amet, consectetur adipisci
ng elit. Aliquam molestie dui et turpis tincidunt bibendum. Sed libero tellus, dapibus et aliquam ac, dignissim eu dui. Ullamcorper lectus lacu
tus finibus gravida. Quisque lacinia viverra blandit. In posuere metus non diam mattis, at condimentum libero facilisis.

Morbi egestas justo sit amet nunc luctus placerat. In consectetur, mi ac rutrum molestie, neque risus pulvinar nisi, hendrerit molestie sem justo s
it amet velit. Maecenas suscipit odio felis, at cursus ligula hendrerit quis. Pellentesque habitant morbi tristique senectus et netus et malesuada
fames ac turpis egestas. Donec efficitur, mi nec iaculis condimentum, nisi nibh tincidunt orci, eget commodo ligula justo in est. Maecenas nec accu
san neque. Interdum et malesuada fames ac ante ipsum primis in faucibus.

mié 04 jul 2018 23:17:57 -63
```

Figura 1.1: Archivo `lorem.ipsium`

Como podemos ver en la imagen, el archivo contiene texto. Hay varios párrafos. Muchas líneas y aún más palabras.

Es importante destacar que *Lorem Ipsum* es un texto que se usa habitualmente en diseño gráfico en demostraciones de tipografías o de borradores de diseño para probar el diseño visual. Nosotros podemos usarlo para probar este tipo de utilidades.

Podríamos contar palabra por palabra para obtener cada uno de los datos mencionados antes, pero sería realmente engorroso (por no mencionar lo tedioso y aburrido). Al margen de que esta tarea no es de lo más interesante, el resultado sí puede ser de mucha utilidad. Para profesionales de la comunicación es importante saber cuántas palabras escriben para un artículo. O cuántos párrafos son los que llevan escritos en un capítulo. Hacerlo a mano sería un castigo para quien tuviera que llevarlo adelante.

Con el comando `wc` en un instante podemos obtener la cantidad de líneas del documento, la cantidad de palabras y la cantidad de bytes. Para el caso de los archivos de texto, la cantidad de bytes se traduce directamente en la cantidad de caracteres. En la figura podemos observar que `wc` nos informa que el archivo `lorem.ipsum` tiene 10 líneas, 508 palabras y 3474 bytes. Cabe destacar que los bytes se traducen a cada uno de los

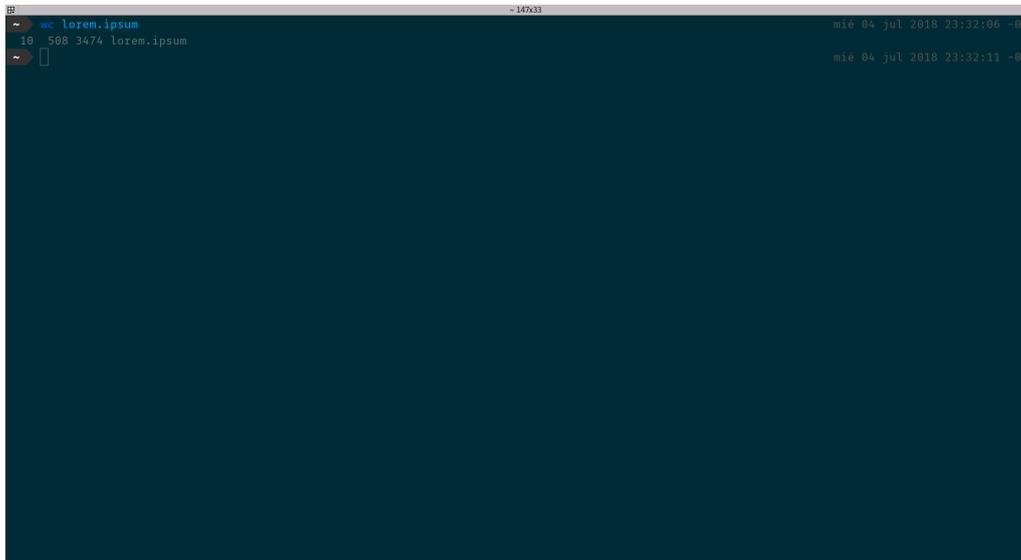
A terminal window with a dark background. The prompt is root@kali:~#. The command 'wc lorem.ipsum' has been executed, and the output is '10 508 3474 lorem.ipsum'. The terminal title bar shows '-107x33'. On the right side, there are two timestamps: 'mie 04 jul 2018 23:32:06 -05' and 'mie 04 jul 2018 23:32:11 -05'.

Figura 1.2: `wc` aplicado a `lorem.ipsum`

caracteres que hay en el archivo, este número incluye los caracteres imprimibles como los no imprimibles. Por eso no hay que traducir este número de caracteres directamente a la cantidad de letras, ya que este valor incluye los espacios y también los saltos de línea.

1.2.1 Página del manual

Se puede ejecutar el comando `man wc` para ver y leer la página del manual de dicho comando con la terminal.

También se encuentra disponible en Internet en <https://linux.die.net/man/1/wc>

Las páginas de manual se pueden navegar con las flechas de dirección del teclado.

Para salir se debe presionar la tecla `q`.

```
HC(1) user Commands ~213x47 wc(1)
NAME
wc - print newline, word, and byte counts for each file

SYNOPSIS
wc [OPTION]... [FILE]...
wc [OPTION]... --files-from-F

DESCRIPTION
Print newline, word, and byte counts for each FILE, and a total line if more than one FILE is specified. A word is a non-zero-length
sequence of characters delimited by white space.

With no FILE, or when FILE is -, read standard input.

The options below may be used to select which counts are printed, always in the following order: newline, word, character, byte, maximum
line length.

-c, --bytes
    print the byte counts

-m, --chars
    print the character counts

-l, --lines
    print the newline counts

--files-from-F
    read input from the files specified by NUL-terminated names in file F; if F is - then read names from standard input

-L, --max-line-length
    print the maximum display width

-w, --words
    print the word counts

--help display this help and exit
--version
    output version information and exit

AUTHOR
Written by Paul Rubin and David Mackenzie.

REPORTING BUGS
GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Manual page wc(1) line 1/57/22% (press h for help or q to quit)
```

Figura 1.3: Página de manual de wc

1.3 Consiga

Se desea que los estudiantes puedan realizar un programa en Python que reproduzca un funcionamiento similar al que tiene el comando wc.

```
HC ~ wc lorem ipsum ~147x33 30s < mié 04 jul 2018 23:41:33 -01
 10 508 3474 lorem ipsum
~ vim miprograma.py ~ jue 05 jul 2018 00:06:38 -01
~ python3 miprograma.py 2m < jue 05 jul 2018 00:09:04 -01
Ingrese el nombre del archivo: lorem ipsum
Resumen del archivo lorem ipsum
El archivo tiene 10 lineas
El archivo tiene 508 palabras
El archivo tiene 3474 caracteres
~ 6s < jue 05 jul 2018 00:09:17 -01
```

Figura 1.4: Python wc

Al ejecutar el archivo con Python3 el programa deberá solicitar al usuario el nombre del archivo del cuál se desea conocer sus estadísticas. Si el nombre de archivo no existe, el programa debe indicarlo con un mensaje claro y amigable para el usuario. Si el archivo existe, deberá imprimir las estadísticas como se muestra en la figura.

1.4 Criterios de aprobación

A continuación se describen los criterios y lineamientos que deberán respetarse en el desarrollo del trabajo.

Los estudiantes deberán ser capaces de responder algunas preguntas:

- ¿Cómo es el flujo del programa que desarrollaron?
- ¿Qué estructura utilizaron? ¿Por qué?
- ¿Cómo se podría mejorar?

1.4.1 Programa

Además de satisfacer la especificación de la consigna, el programa debe cumplir con los siguientes requerimientos:

- No debe fallar ninguna prueba.
- Debe estar adecuadamente estructurado y modularizado, utilizando funciones definidas en forma lo más genérica posible.
- El código debe ser claro y legible.
- Todas las funciones deben estar adecuadamente documentadas, y donde sea necesario el código debe estar acompañado de comentarios.

1.4.2 Examen

Para la fecha de presentación del examen los estudiantes deberán venir con una copia impresa del código fuente que desarrollaron y enviarlo previamente por correo electrónico a colomboleandro@ifts18.edu.ar

Durante la mesa de examen el docente indicará a cada estudiante una modificación y/o agregado que deberán hacer en su programa en la copia impresa que trajeron del mismo.

Los estudiantes deberán minimizar las modificaciones a su programa maximizando la utilización de su propio código. Es decir, deberán reescribir la menor cantidad de sentencias posibles.

El docente podrá solicitar a los estudiantes que respondan preguntas oralmente para cumplimentar o despejar dudas respecto de su implementación de solución durante la mesa de examen.

Capítulo 2

Recursos y sugerencias

2.1 Lorem Ipsum

En Internet se pueden encontrar varios sitios que generan texto Lorem Ipsum para que puedan probar su software. Por ejemplo:

<https://www.lipsum.com/>

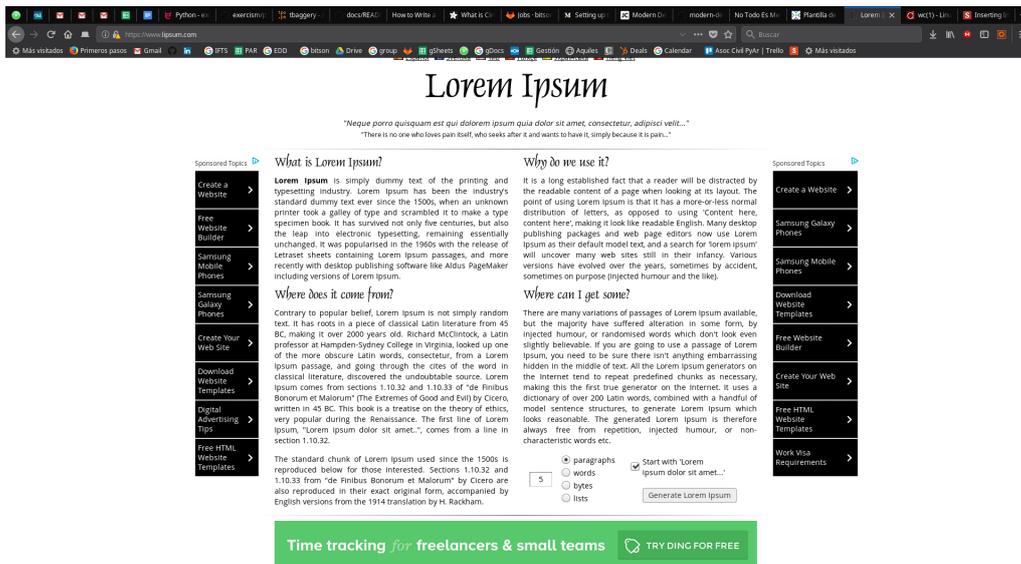


Figura 2.1: Página generadora de texto Lorem Ipsum

En este sitio van a poder definir qué tipo de texto quieren generar y una vez generado obtienen los valores estadísticos que les servirán para validar el correcto funcionamiento de su versión de wc.