university
of
tyumen

school
of advanced
studies

Information Technology - advanced

Lecture 2
Logical representation of numbers

Fabio Grazioso - *April 2018*

Todays’ lecture

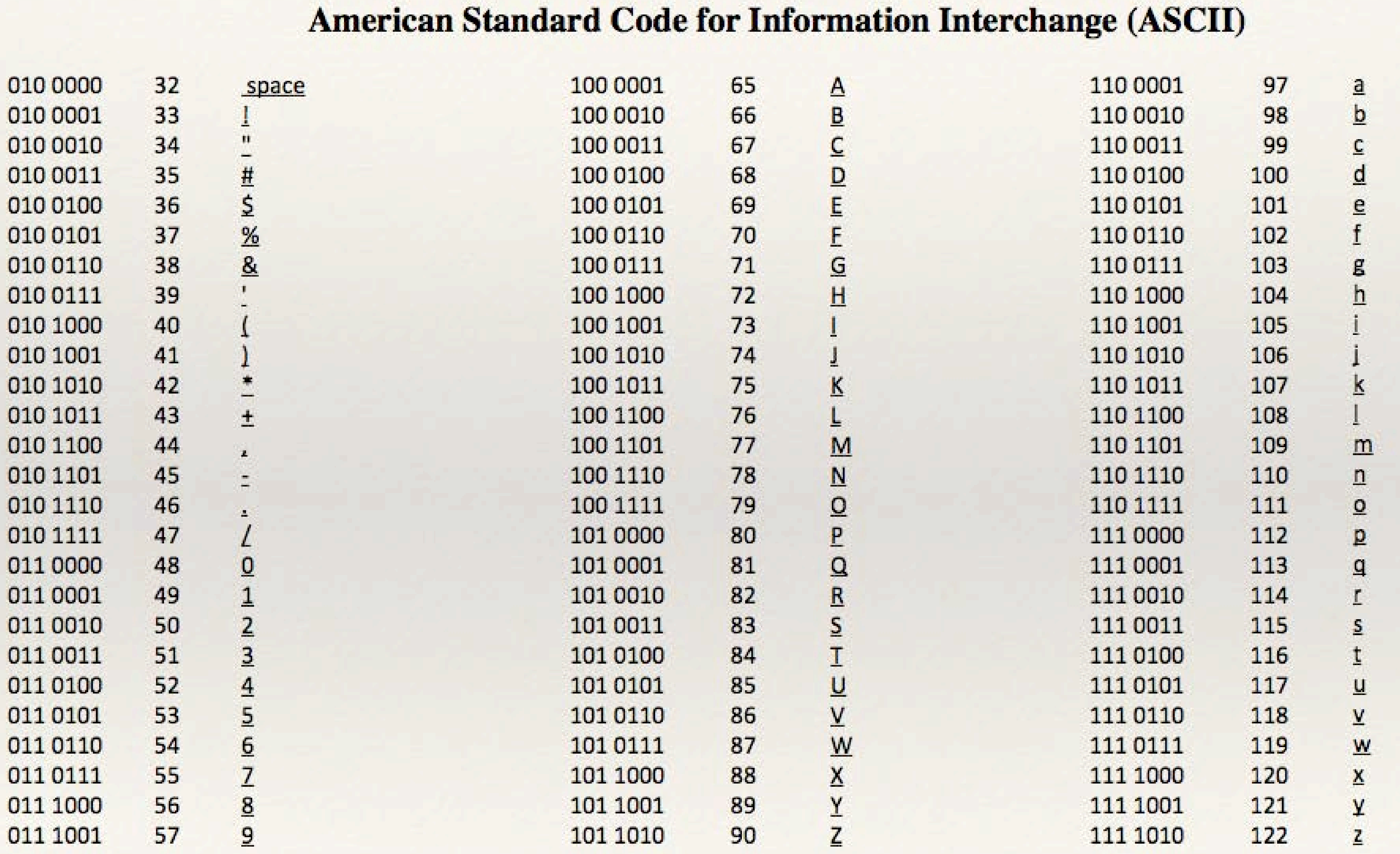
Computer problem-solving

Summary of the lecture

- Text encodings
- General definitions of Operative Systems
- Discussion of the different User Interfaces
- Command Line Interfaces
- Graphical User Interfaces
- Software interfaces in popular culture
- Programming environment
    - IDE
    - debugger
    - compiler
    - interpreter

Text encodings

ascii characters



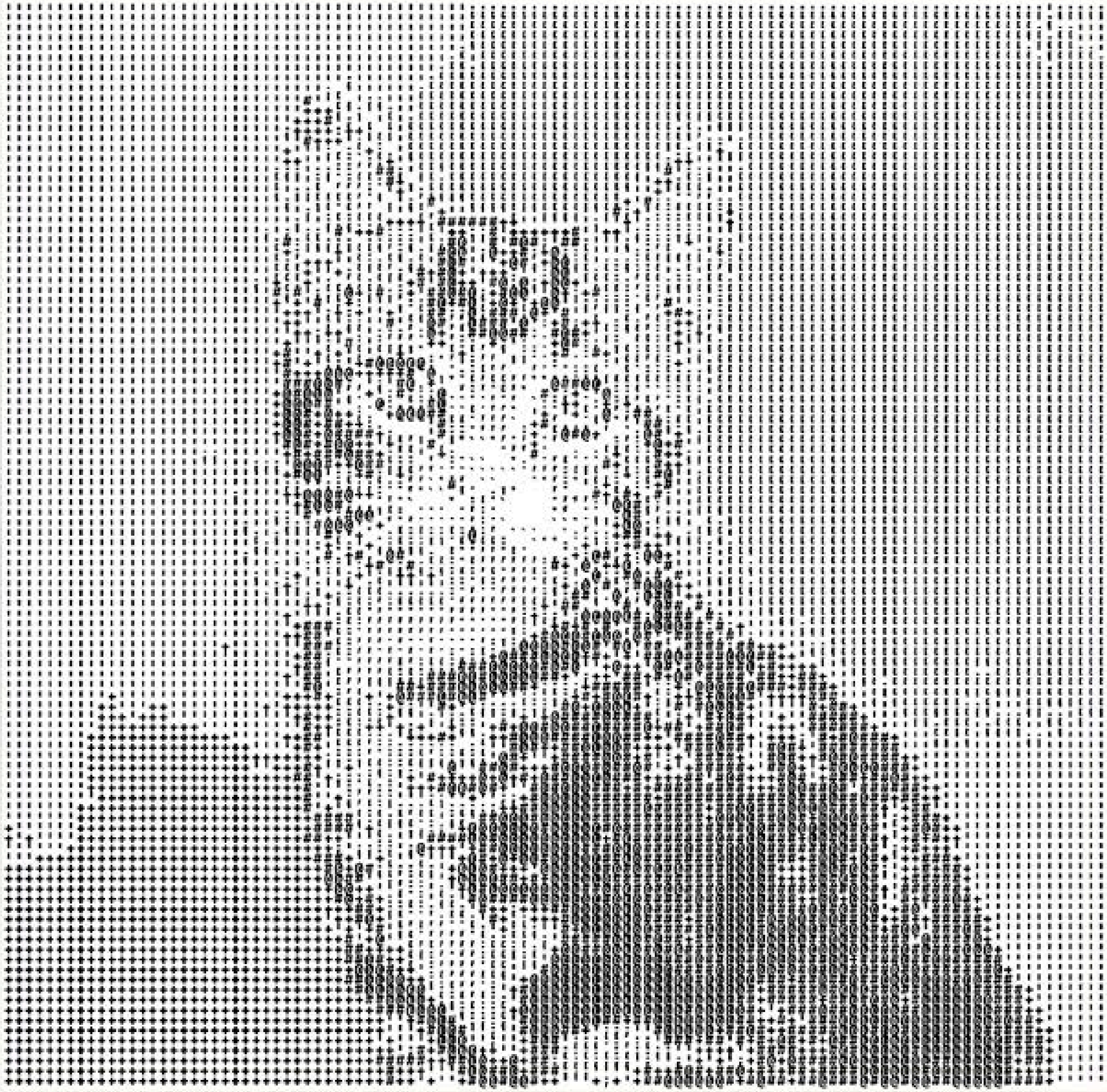
American Standard Code for Information Interchange (ASCII)

010 0000	32	space	100 0001	65	A	110 0001	97	a
010 0001	33	!	100 0010	66	B	110 0010	98	b
010 0010	34	"	100 0011	67	C	110 0011	99	c
010 0011	35	#	100 0100	68	D	110 0100	100	d
010 0100	36	$	100 0101	69	E	110 0101	101	e
010 0101	37	%	100 0110	70	F	110 0110	102	f
010 0110	38	&	100 0111	71	G	110 0111	103	g
010 0111	39	'	100 1000	72	H	110 1000	104	h
010 1000	40	(	100 1001	73	I	110 1001	105	i
010 1001	41	)	100 1010	74	J	110 1010	106	j
010 1010	42	*	100 1011	75	K	110 1011	107	k
010 1011	43	+	100 1100	76	L	110 1100	108	l
010 1100	44	,	100 1101	77	M	110 1101	109	m
010 1101	45	-	100 1110	78	N	110 1110	110	n
010 1110	46	.	100 1111	79	O	110 1111	111	o
010 1111	47	/	101 0000	80	P	111 0000	112	p
011 0000	48	0	101 0001	81	Q	111 0001	113	q
011 0001	49	1	101 0010	82	R	111 0010	114	r
011 0010	50	2	101 0011	83	S	111 0011	115	s
011 0011	51	3	101 0100	84	T	111 0100	116	t
011 0100	52	4	101 0101	85	U	111 0101	117	u
011 0101	53	5	101 0110	86	V	111 0110	118	v
011 0110	54	6	101 0111	87	W	111 0111	119	w
011 0111	55	7	101 1000	88	X	111 1000	120	x
011 1000	56	8	101 1001	89	Y	111 1001	121	y
011 1001	57	9	101 1010	90	Z	111 1010	122	z

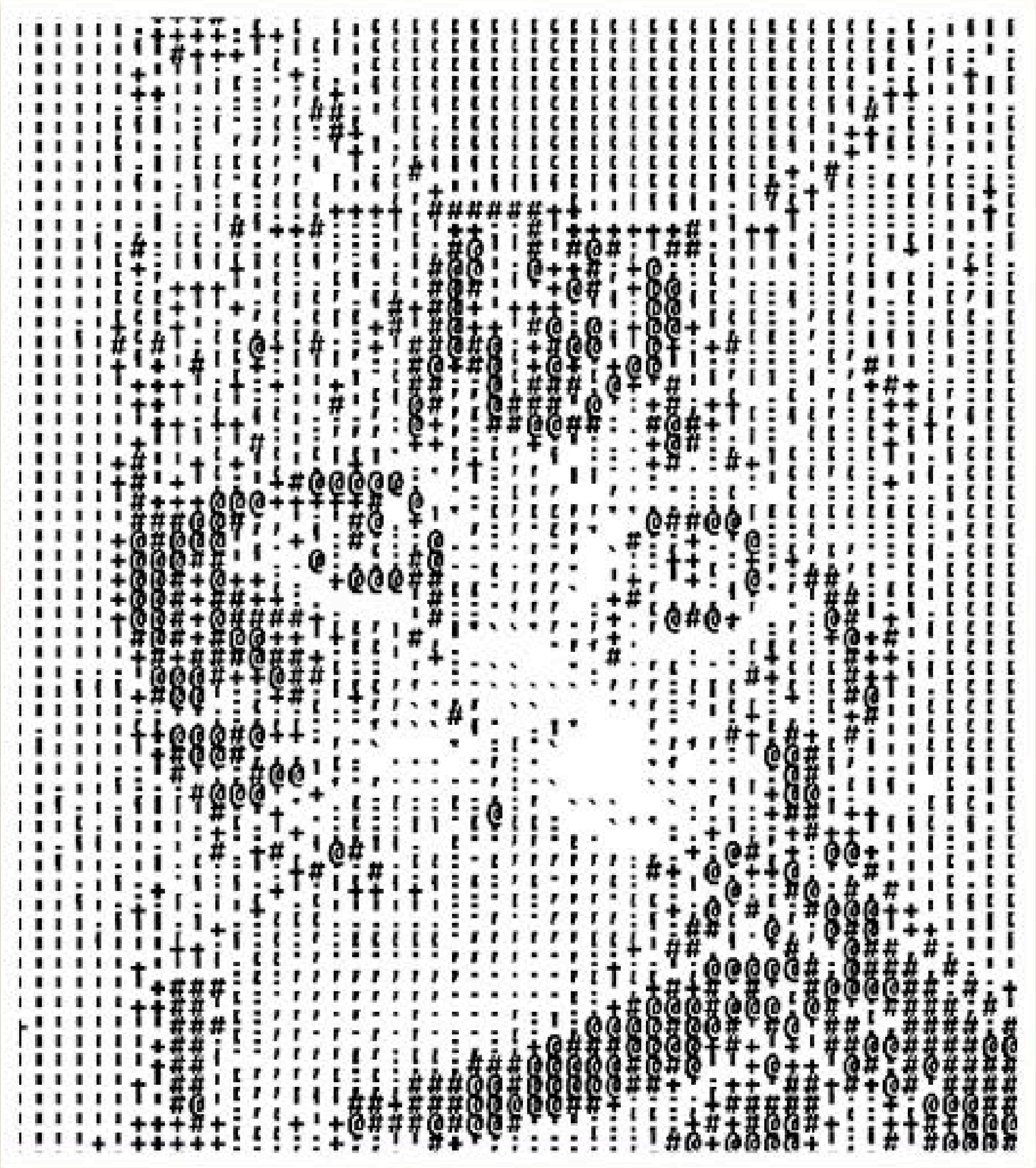
ascii characters

```
                        ,.                .,
                      ,: ':.    .,.    .:' :,
                      ,',   '.:'   ':.'   ,',
                      : '.  '         '  .' :
                      ', : '           ' : ,'
                       '.' .,:,.   .,:,. '.'
                        ,:   V '. .' V    :,
                      ,:        / '        :,
                      ,:                   :,
                        ,:      =:=       :,
                         ,: ,     :     , :,
                          :' ',.,' ',.,:' ':
                         :'     ':WW::'   '.
                        .:'      '::::'   ':
                       ,:        '::::'    :,
                       :'         ':::'    ':
                      ,:          ':''     :.
                    .:'             '.     ',.
                   ,:'               ''     '.
                   .:'               .',    ':
                  .:'               .'.,     :
                  .:                .,''     :
                  ::                .,''    ,:
                  ::              .,'','   .:'
               .,::'.           .,','     ::::.
             .:'     ',.       ,:,       ,WWWWW,
      :'        :       :W:'     :WWWWWWW,          .,.
      :         ',     WWW      WWWWWWWWW          '::,
     '.         ',    WWW     :WWWWWWWWW            '::,
      '.         :    WWW     :WWWWWWWW'             :::
       '.       ,:    WWW     :WWWWWWW'             .:::
        '.     .W:     WWW     :WWWWWW'           .,:::'
          '.   :WW:     WWW     :WWWWW'      .,,:::::''
            .,'   ''::     :W:     :WWWWW.  .,::::''
           ,'        ''','',',',''','''WWWWW::::''
              ':,,,,,,,':  :  : : :  :  :WWWW'''
```

ascii characters



ascii characters



ascii encoding

- The best known and most widely used character encoding standard is the **American Standard Code for Information Interchange** (ASCII).
- The first version of ASCII was published in **1964** as a standard way of representing textual data in computer memory and sending it over communication links between computers.
- ASCII is based on a **seven-bit byte.** Each byte represented a character, and characters were represented by assigning them to individual binary numbers.

Gillam p7

ascii encoding

what is the highest value that we can write with 7 binary digits?

$$\begin{array}{ccccccc} 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}$$

$$\begin{array}{cccccccc} 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

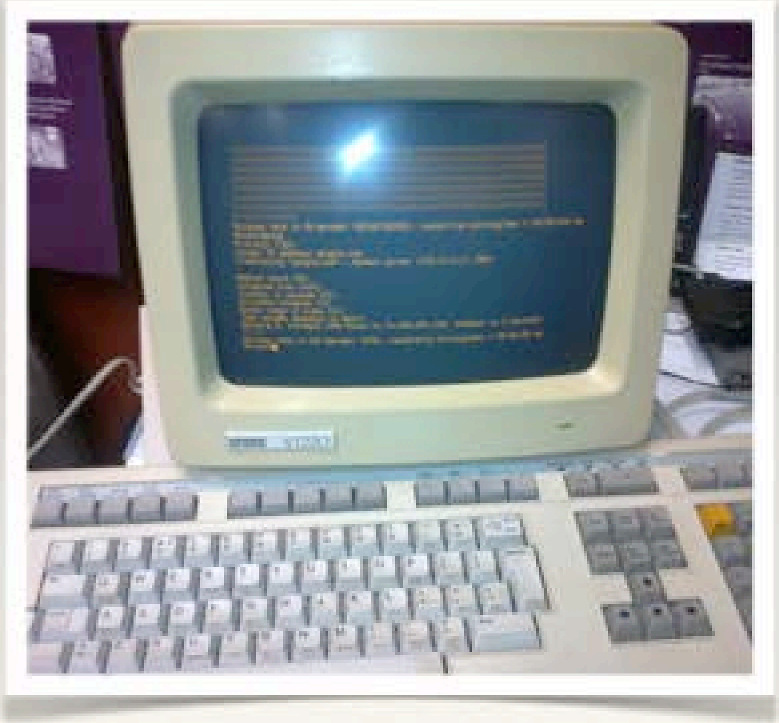
$$2^7 = 128$$

Gillam p7

ascii encoding

- Perhaps the main deficiency in ASCII comes from the "A" in its name: **American**. ASCII is an American standard, and was designed for the storage and transmission of English text. 95 characters are sufficient for representing English text, barely, but that's it. On early teletype machines, ASCII could also be used to represent the accented letters found in many European languages, but this capability disappeared in the transition from teletypes to CRT terminals.

CRT = Cathode Ray Tube



Gillam p7

ascii encoding

- Perhaps the main deficiency in ASCII comes from the "A" in its name: **American**. ASCII is an American standard, and was designed for the storage and transmission of English text. 95 characters are sufficient for representing English text, barely, but that's it. On early teletype machines, ASCII could also be used to represent the accented letters found in many European languages, but this capability disappeared in the transition from teletypes to CRT terminals.
- So, as computer use became more and more widespread in different parts of the world, alternative methods of representing characters in computers arose for representing other languages, leading to the situation we have today, where there are generally three or four different encoding schemes for every language and writing system in use today.

Gillam p7

Unicode

- Unicode is the latest of several attempts to solve this Tower of Babel problem by creating a universal character encoding.
- Its main way of doing this is to increase the size of the possible encoding space by **increasing the number of bits used to encode each character**.
- Most other character encodings are based upon an **eight-bit byte**, which provides enough space to encode a maximum of **256 characters** (in practice, most encodings reserve some of these values for control signals and encode fewer than 256 characters).

Gillam p7

Unicode

- Unicode uses a 16-bit word to encode characters, allowing up to 65,536 characters to be encoded. 65,000 characters, with careful management, is enough to allow encoding of the vast majority of characters in the vast majority of written languages in use today.
- The current version of Unicode, version 3.2, actually encodes 95,156 different characters—it actually does use a scheme to represent the less-common characters using two 16-bit units, but with 50,212 characters actually encoded using only a single unit, you rarely encounter the two-unit characters. In fact, these 50,212 characters include all of the characters representable with all of the other character encoding methods that are in reasonably widespread use.

Gillam p7

UTF-8

- UTF-8 is a variable width character encoding capable of encoding all 1,112,064 valid code points in **Unicode** using one to four 8-bit **bytes**.
- The name is derived from Unicode Transformation Format – 8-bit.

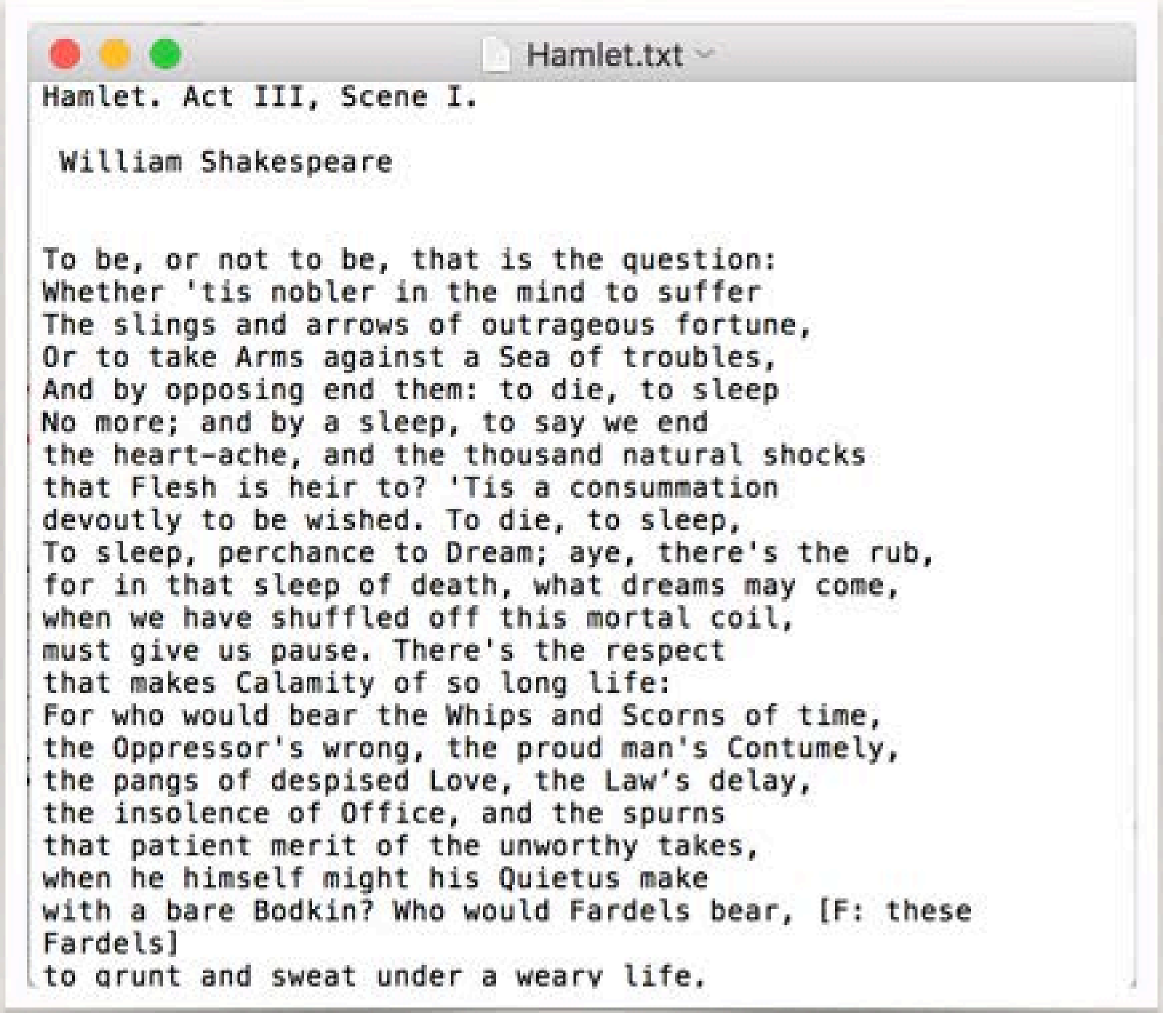
Wikipedia

cyrillic

0410	А	0430	а		
0411	Б	0431	б		
0412	В	0432	в		
0413	Г	0433	г		
0414	Д	0434	д		
0415	Е	0435	е		
0416	Ж	0436	ж		
0417	З	0437	з		
0418	И	0438	и		
0419	Й	0439	й		
041A	К	043A	к		
041B	Л	043B	л		
041C	М	043C	м		
041D	Н	043D	н		
041E	О	043E	о		
041F	П	043F	п		

0420	Р	0440	р
0421	С	0441	с
0422	Т	0442	т
0423	У	0443	у
0424	Ф	0444	ф
0425	Х	0445	х
0426	Ц	0446	ц
0427	Ч	0447	ч
0428	Ш	0448	ш
0429	Щ	0449	щ
042A	Ъ	044A	ъ
042B	Ы	044B	ы
042C	Ь	044C	ь
042D	Э	044D	э
042E	Ю	044E	ю
042F	Я	044F	я

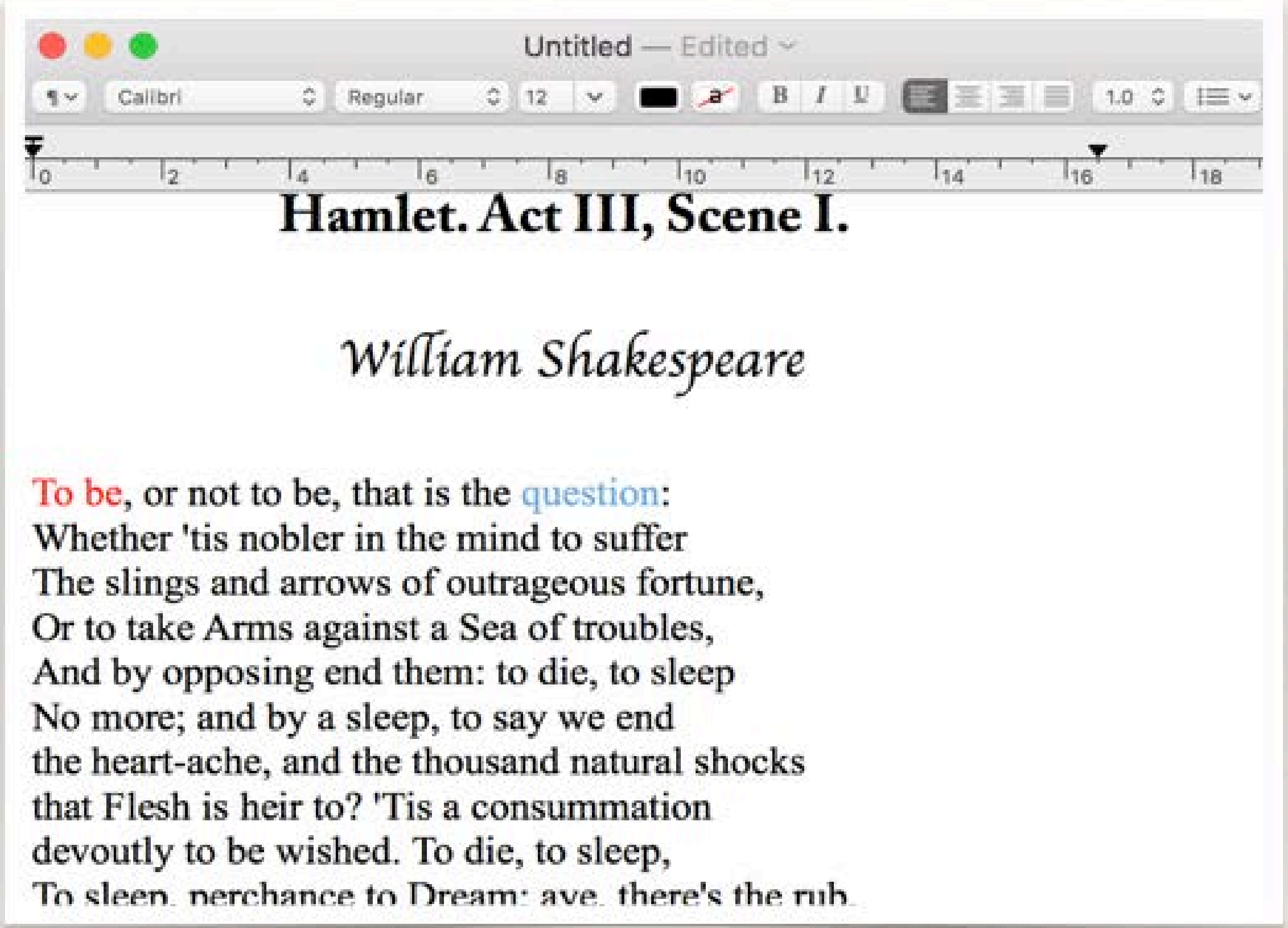
.txt vs .rtf



Hamlet. Act III, Scene I.

William Shakespeare

To be, or not to be, that is the question:
Whether 'tis nobler in the mind to suffer
The slings and arrows of outrageous fortune,
Or to take Arms against a Sea of troubles,
And by opposing end them: to die, to sleep
No more; and by a sleep, to say we end
the heart-ache, and the thousand natural shocks
that Flesh is heir to? 'Tis a consummation
devoutly to be wished. To die, to sleep,
To sleep, perchance to Dream; aye, there's the rub,
for in that sleep of death, what dreams may come,
when we have shuffled off this mortal coil,
must give us pause. There's the respect
that makes Calamity of so long life:
For who would bear the Whips and Scorns of time,
the Oppressor's wrong, the proud man's Contumely,
the pangs of despised Love, the Law's delay,
the insolence of Office, and the spurns
that patient merit of the unworthy takes,
when he himself might his Quietus make
with a bare Bodkin? Who would Fardels bear, [F: these Fardels]
to grunt and sweat under a weary life,



Hamlet. Act III, Scene I.

William Shakespeare

To be, or not to be, that is the question:
Whether 'tis nobler in the mind to suffer
The slings and arrows of outrageous fortune,
Or to take Arms against a Sea of troubles,
And by opposing end them: to die, to sleep
No more; and by a sleep, to say we end
the heart-ache, and the thousand natural shocks
that Flesh is heir to? 'Tis a consummation
devoutly to be wished. To die, to sleep,
To sleep, perchance to Dream; aye, there's the rub,

Rich Text Format

- It's also important to keep in mind what Unicode isn't. First, Unicode is a standard scheme for representing plain text in computers and data communication. It is not a scheme for representing rich text (sometimes called "fancy text" or "styled text"). This is an important distinction. Plain text is the words, sentences, numbers, and so forth themselves. Rich text is plain text plus information about the text, especially information on the text's visual presentation (e.g., the fact that a given word is in italics), the structure of a document (e.g., the fact that a piece of text is a section header or footnote), or the language (e.g., the fact that a particular sentence is in Spanish). Rich text may also include non-text items that travel with the text, such as pictures.
- It can be somewhat tough to draw a line between what qualifies as plain text, and therefore should be encoded in Unicode, and what's really rich text. In fact, debates on this very subject flare up from time to time in the various Unicode discussion forums. The basic rule is that plain text contains all of the information necessary to carry the semantic meaning of the text—the letters, spaces, digits, punctuation, and so forth. If removing it would make the text unintelligible, then it's plain text.

Gillam p7

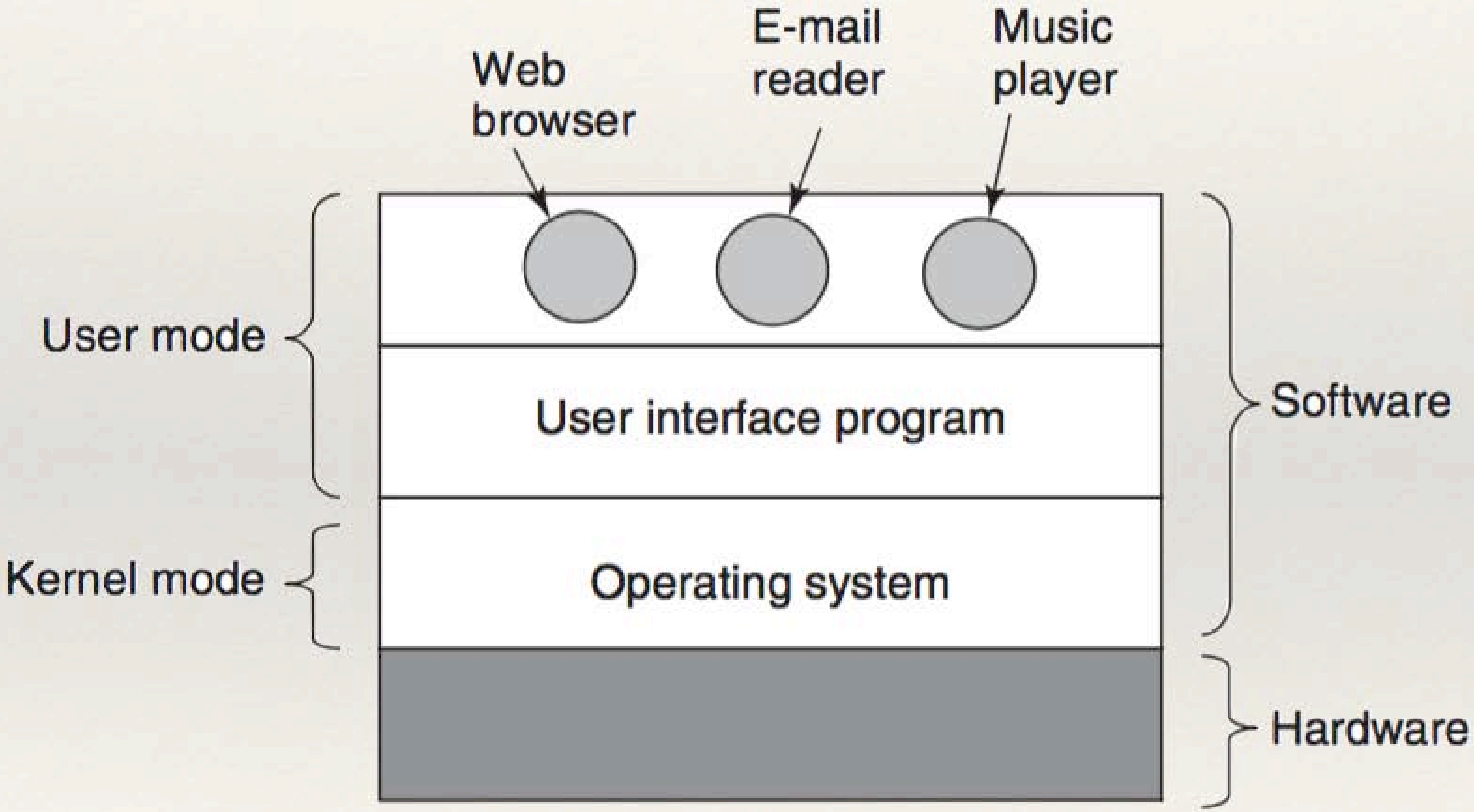
Operative Systems

structure

- An operating system acts as an intermediary between the user of a computer and the computer hardware.
- The purpose of an operating system is to provide an environment in which a user can execute programs in a convenient and efficient manner.

📖 Silberschatz p1

structure



Tanenbaum p2

OS abstraction

- The **architecture** of most computers at the machine-language level is primitive and awkward to program, especially for input/output.
- To give an example, consider modern **SATA** (Serial ATA) **hard disks** used on most computers.
- A book (Anderson, 2007) describing an early version of the interface to the disk (the details for a programmer to use the disk) was more than 450 pages long.
- Clearly, no sane programmer would want to deal with this disk at the hardware level.
- Instead, a piece of software, called a **disk driver**, deals with the hardware and provides an interface to read and write disk blocks, without getting into the details.
- Operating systems contain many **drivers** for controlling I/O devices.
- But even this level is much too low for most applications.
- For this reason, all operating systems provide **another layer of abstraction for using disks**: files.
- Using this abstraction, programs can create, write, and read **files**, without having to deal with the messy details of how the hardware actually works.
- This **abstraction** is the **key** to managing all this **complexity**.

Tanenbaum p4

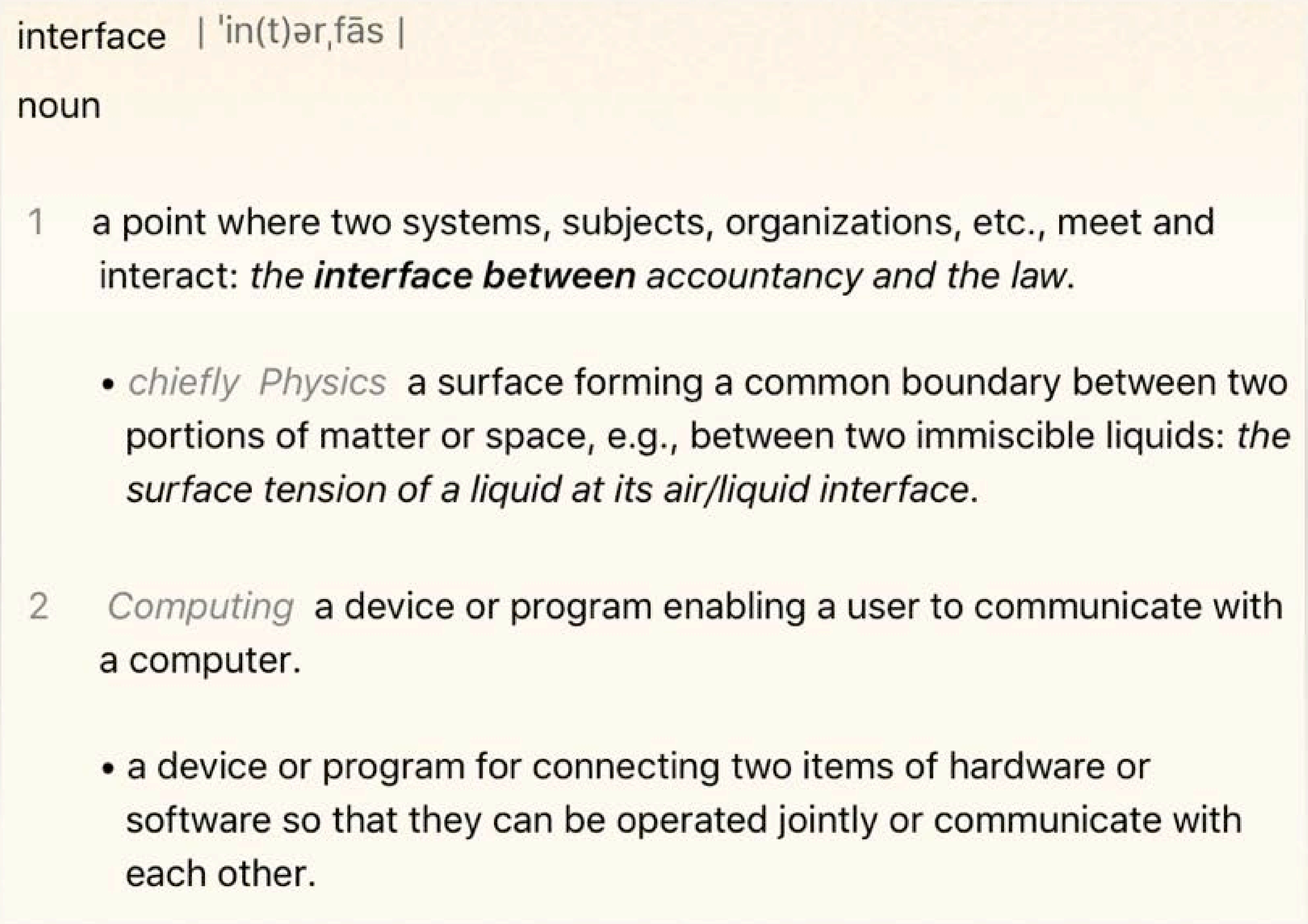
Interfaces

Introduction

UI and GUI

- The Operative System (OS) is the software that lets the user interact with the hardware of a computer.
- How do we communicate with the OS?
- We need an **interface**, something between the user and the OS.

UI and GUI



interface | ˈin(t)ərˌfās |

noun

1 a point where two systems, subjects, organizations, etc., meet and interact: *the **interface between** accountancy and the law.*

- *chiefly Physics* a surface forming a common boundary between two portions of matter or space, e.g., between two immiscible liquids: *the surface tension of a liquid at its air/liquid interface.*

2 *Computing* a device or program enabling a user to communicate with a computer.

- a device or program for connecting two items of hardware or software so that they can be operated jointly or communicate with each other.

UI and GUI

- The generic name for this is the User Interface (UI)
- Historically, the first UIs were commands written in a keyboard, and output written on a printer

CLI

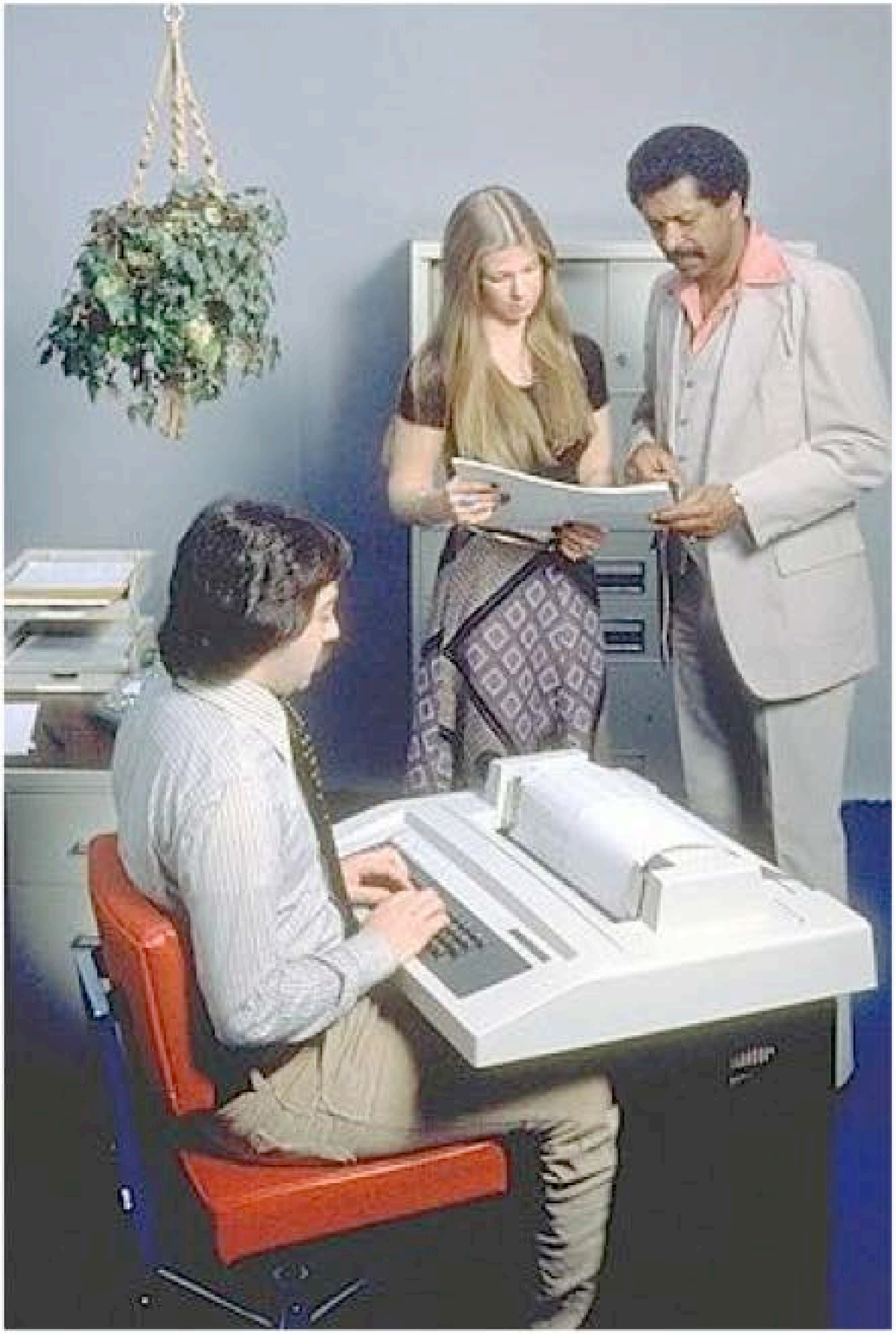
Digital DECWriter II

The LA36 DECwriter II was the company's first commercially successful keyboard terminal and became the de facto market standard.

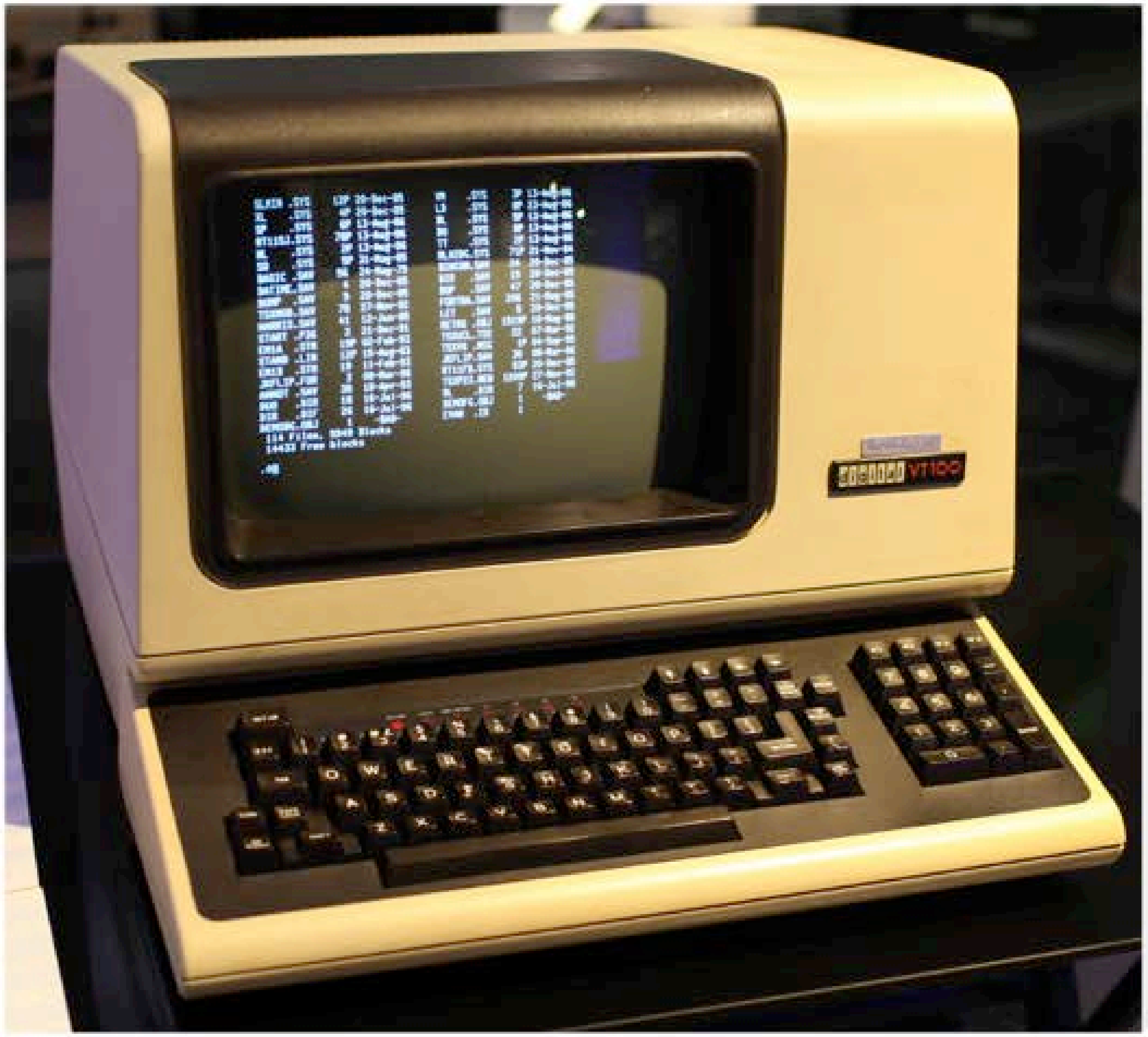
The LA-36 utilises all solid state logic and has an electronic keyboard.

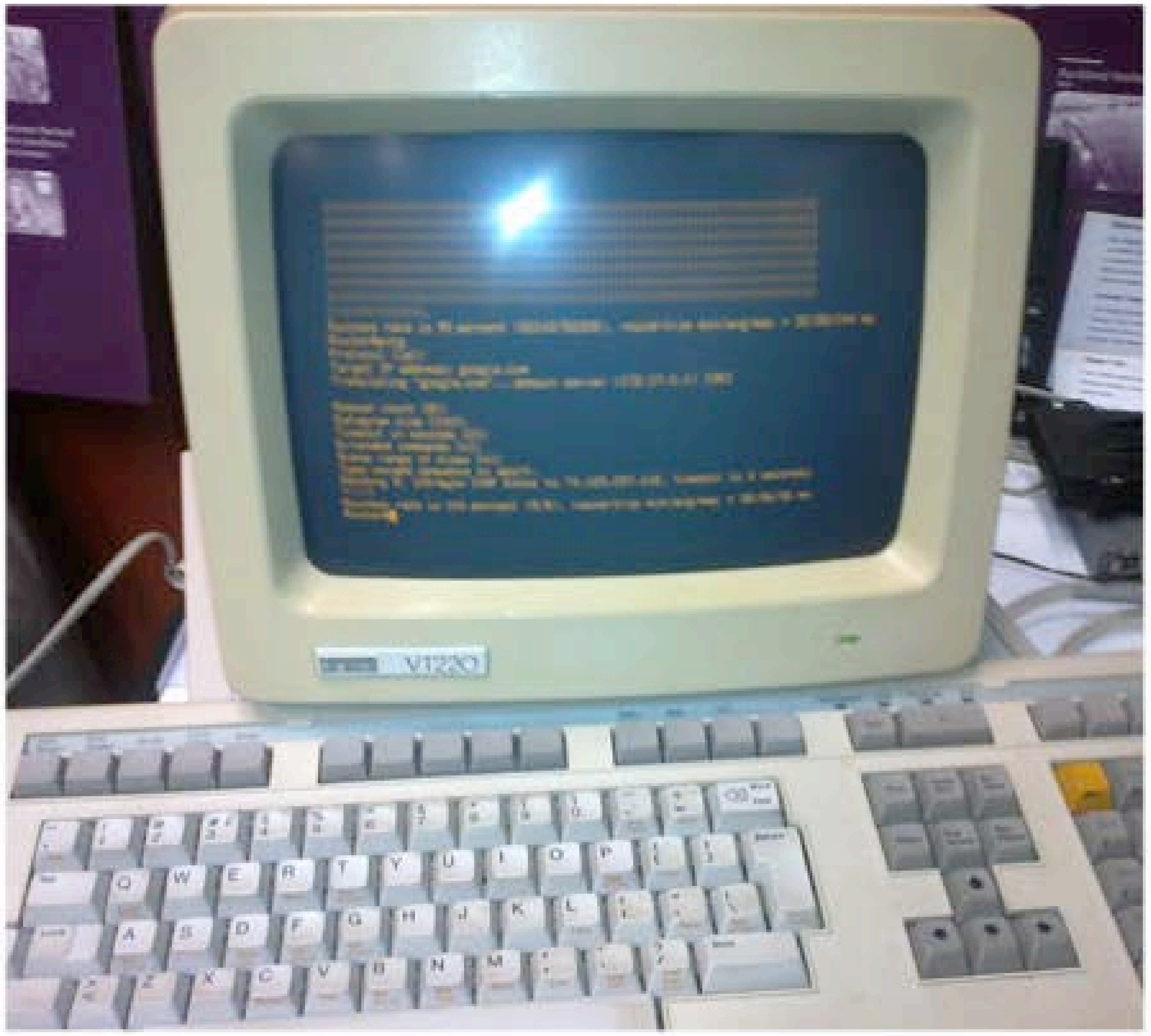
The printer mechanism uses a dot-matrix technique to print 132 columns of text across standard 14 inch computer forms.
Normal printing speed is 30 characters per second using a communications rate of 300 bps.
LA-36 is capable of printing at a catch-up speed of 60 cps for short periods. Some LA-36es were modified to allow printing at the 60 cps full- time, but in doing this fill characters or delays were required to compensate for print- head return time.

Manufacturer: DEC
Date: 1978



VT100 & VT220 terminals



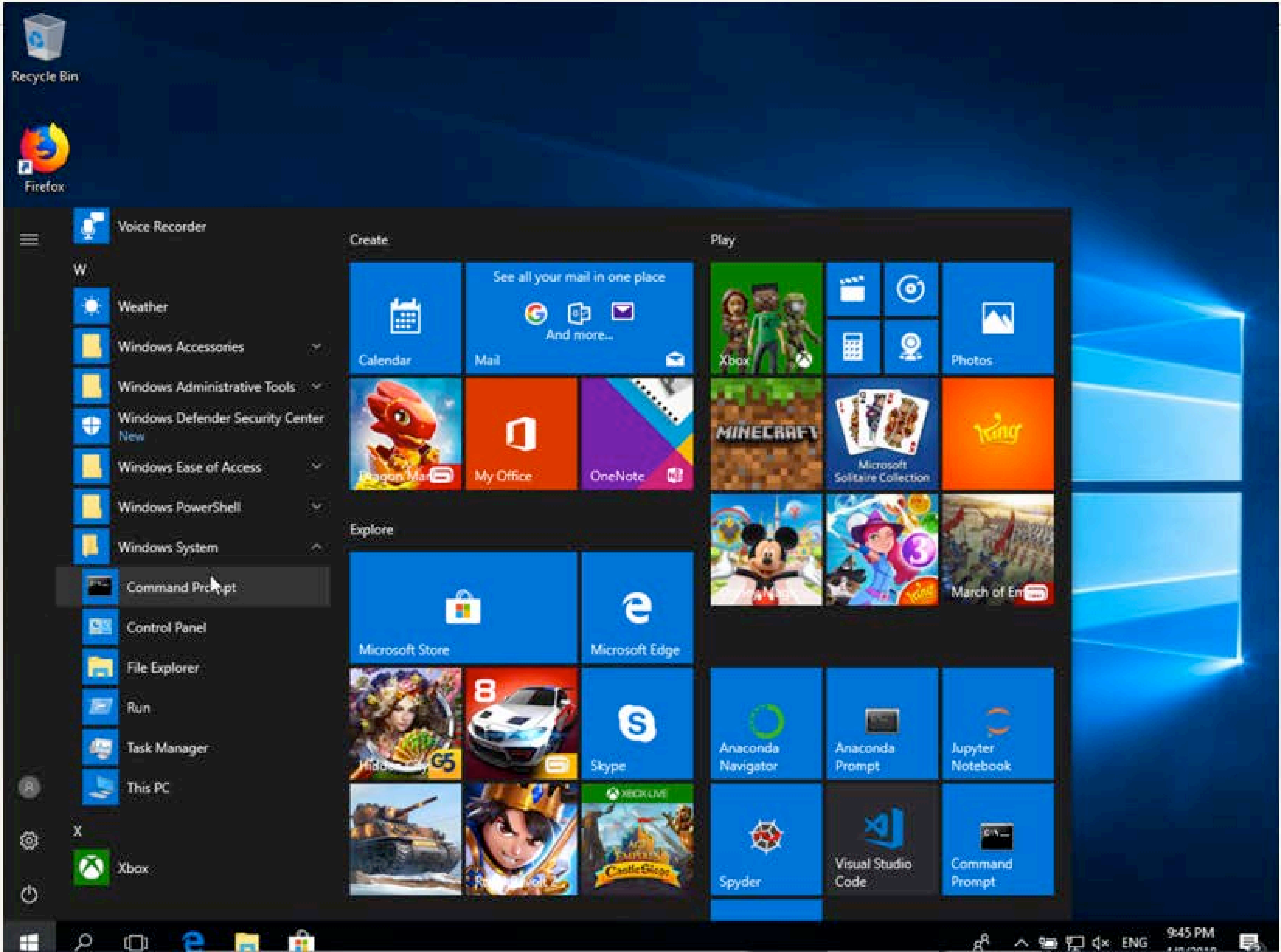


command-line interface (CLI)

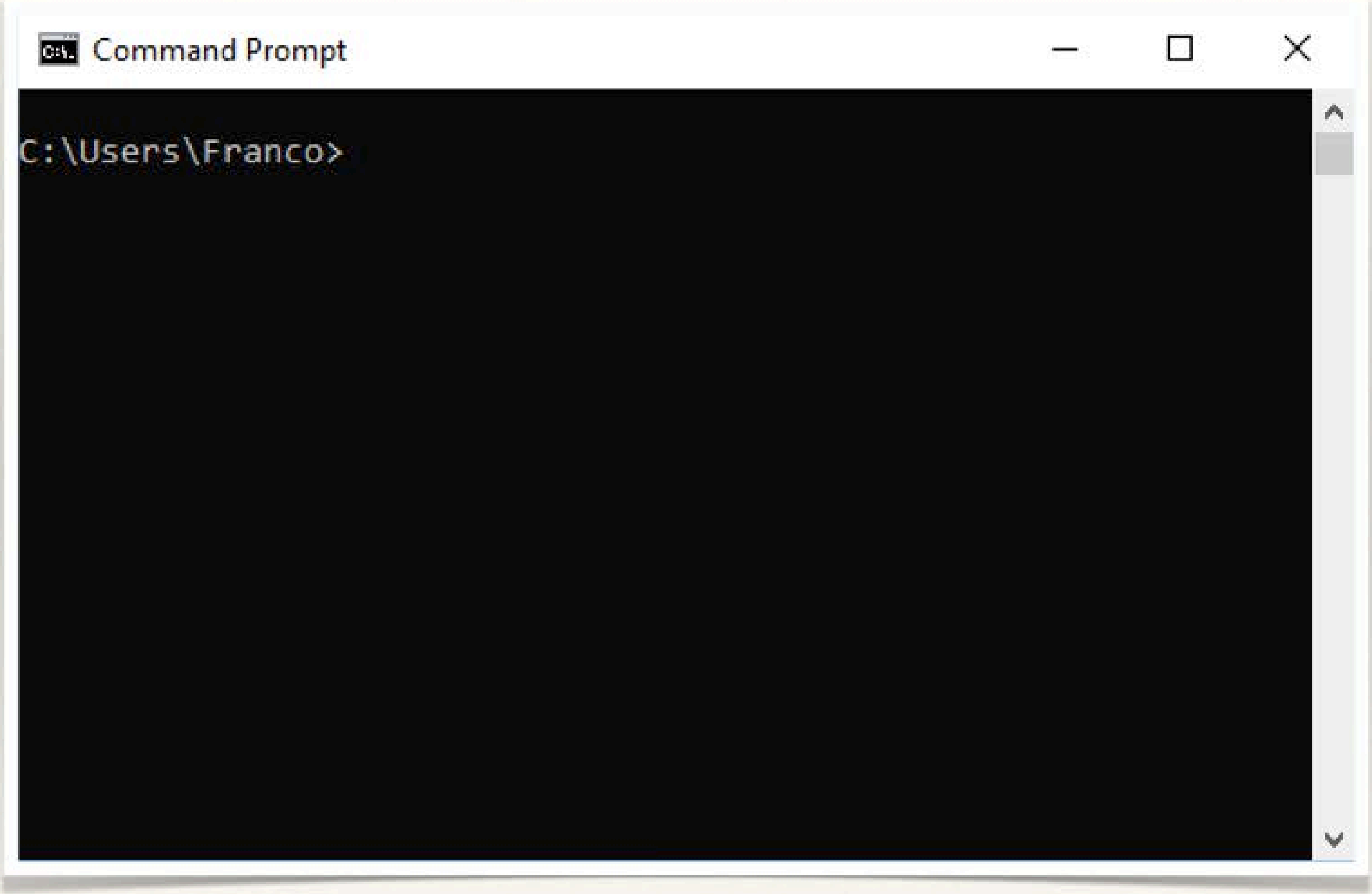
- command-line interface (CLI) has been the typical UI.
- Some well known OS based on CLI were:
  - VAX/VMS (Digital)
  - MS-DOS (Microsoft)
  - ProDOS (Apple)

Windows

Microsoft Windows



Windows Console

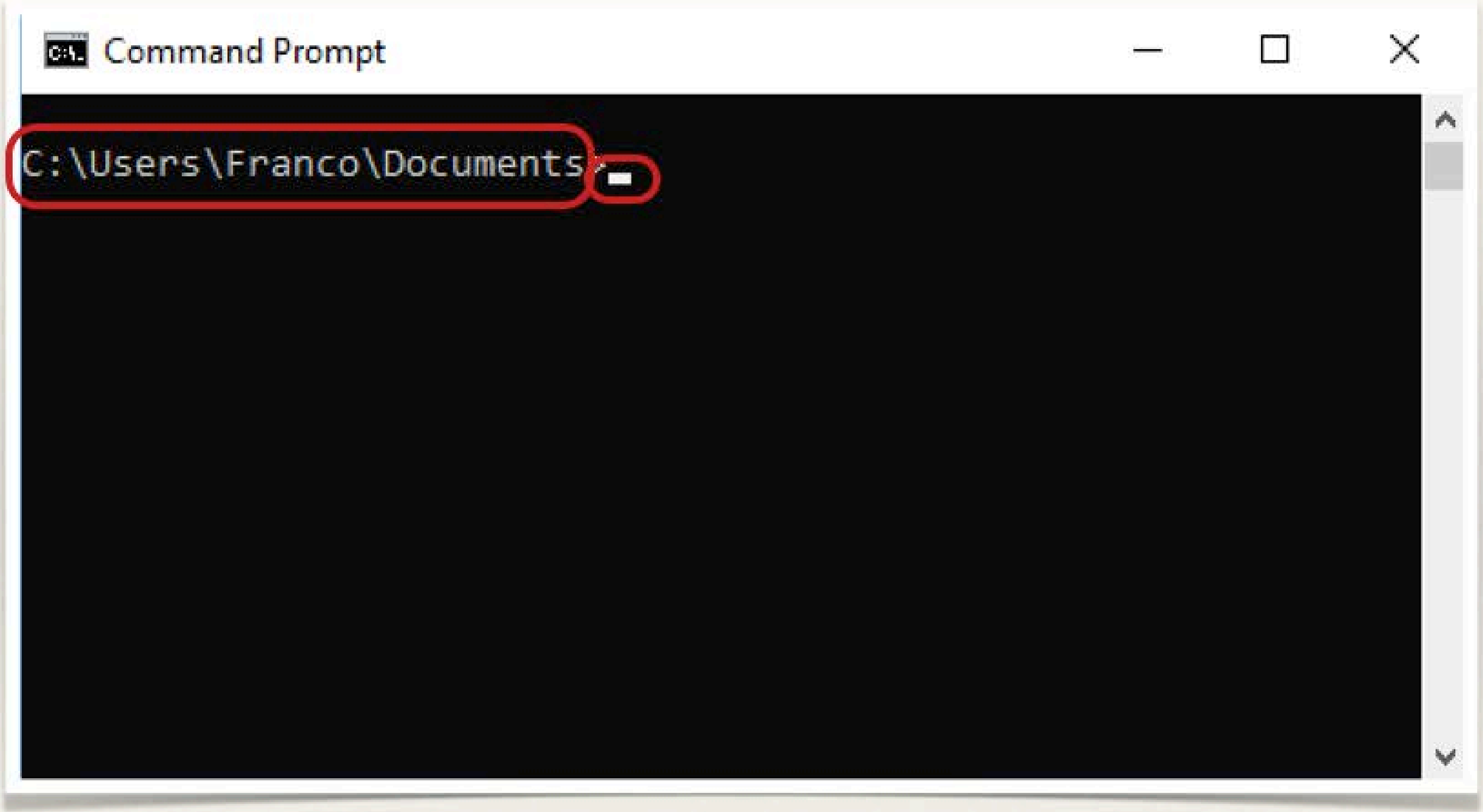


```
C:\Users\Franco>
```

Windows Console

- Working directory

Windows Console



```
C:\Users\Franco\Documents>
```

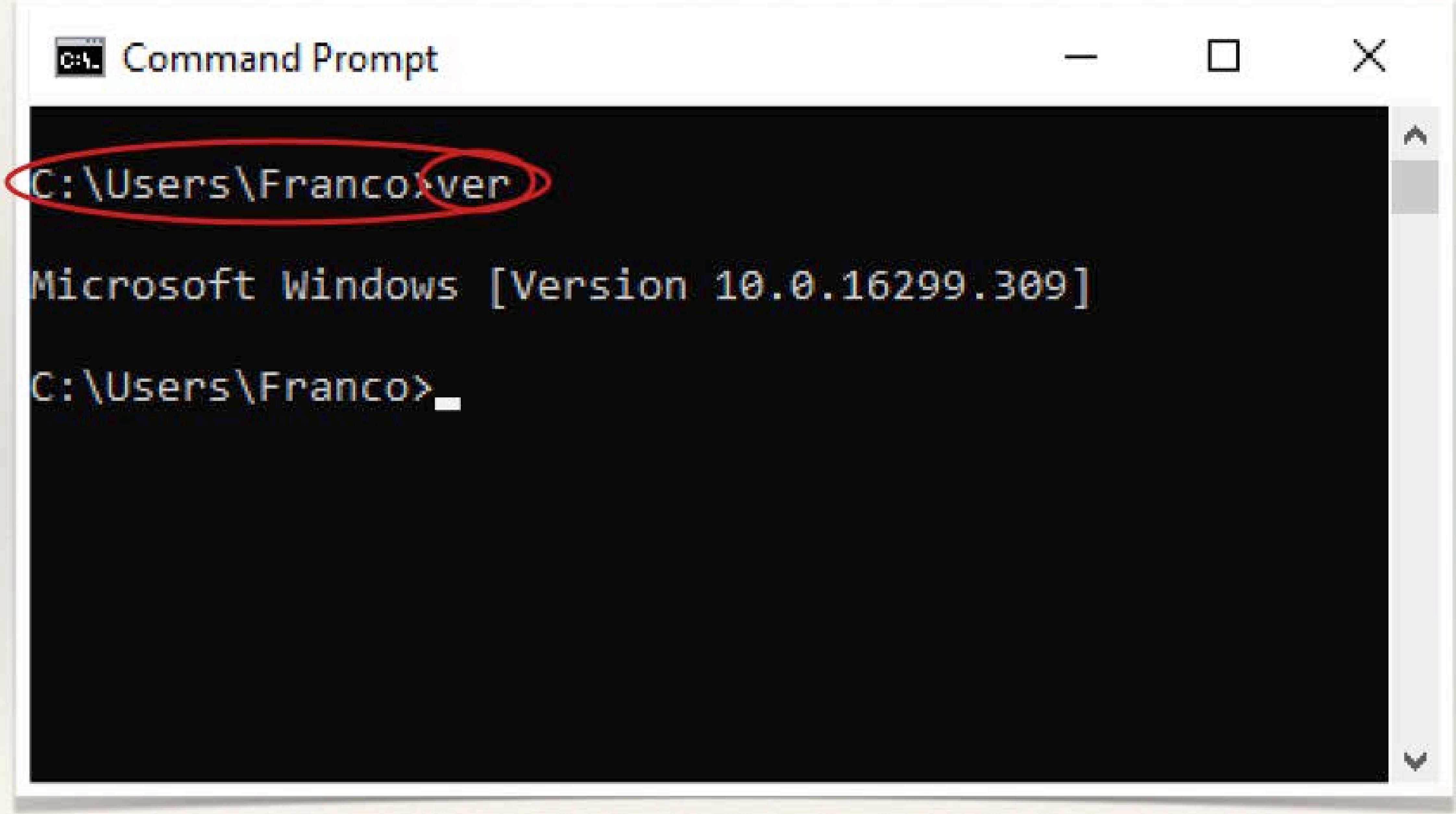
Windows Console

Determine the operating System Version

- Not every version of the command processor supports every command and utility. Consequently, you often need to know which version of the command processor is present on the user's machine. To perform this task, type Ver and press Enter. You'll see an operating system version number, such as Microsoft Windows [Version 6.1.7600], which indirectly tells you which version of the command processor is installed. (The command prompt also displays the version number automatically when you open the window.)

Mueller p16

Windows Console

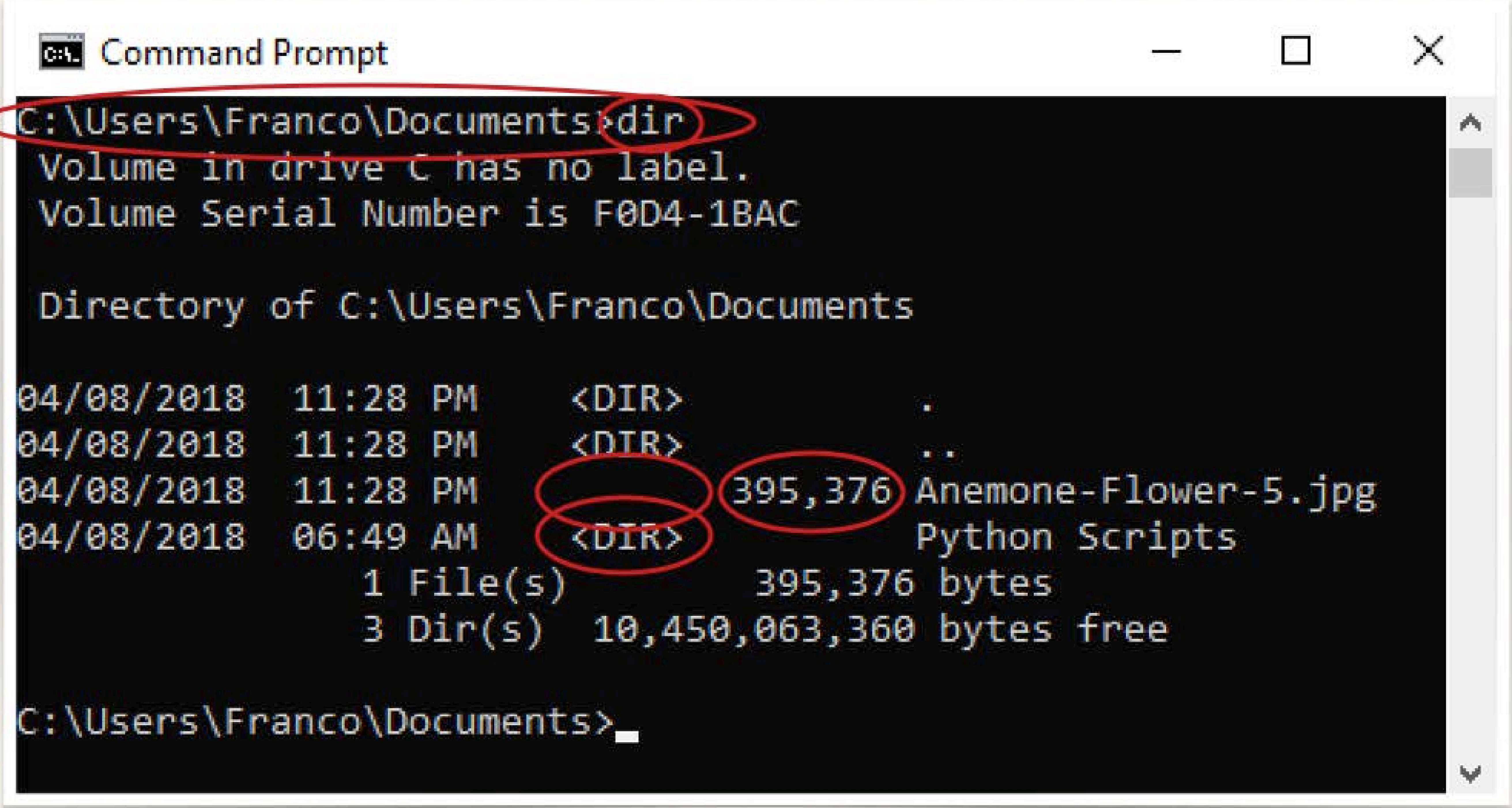


```
C:\Users\Franco>ver

Microsoft Windows [Version 10.0.16299.309]

C:\Users\Franco>_
```

Windows Console

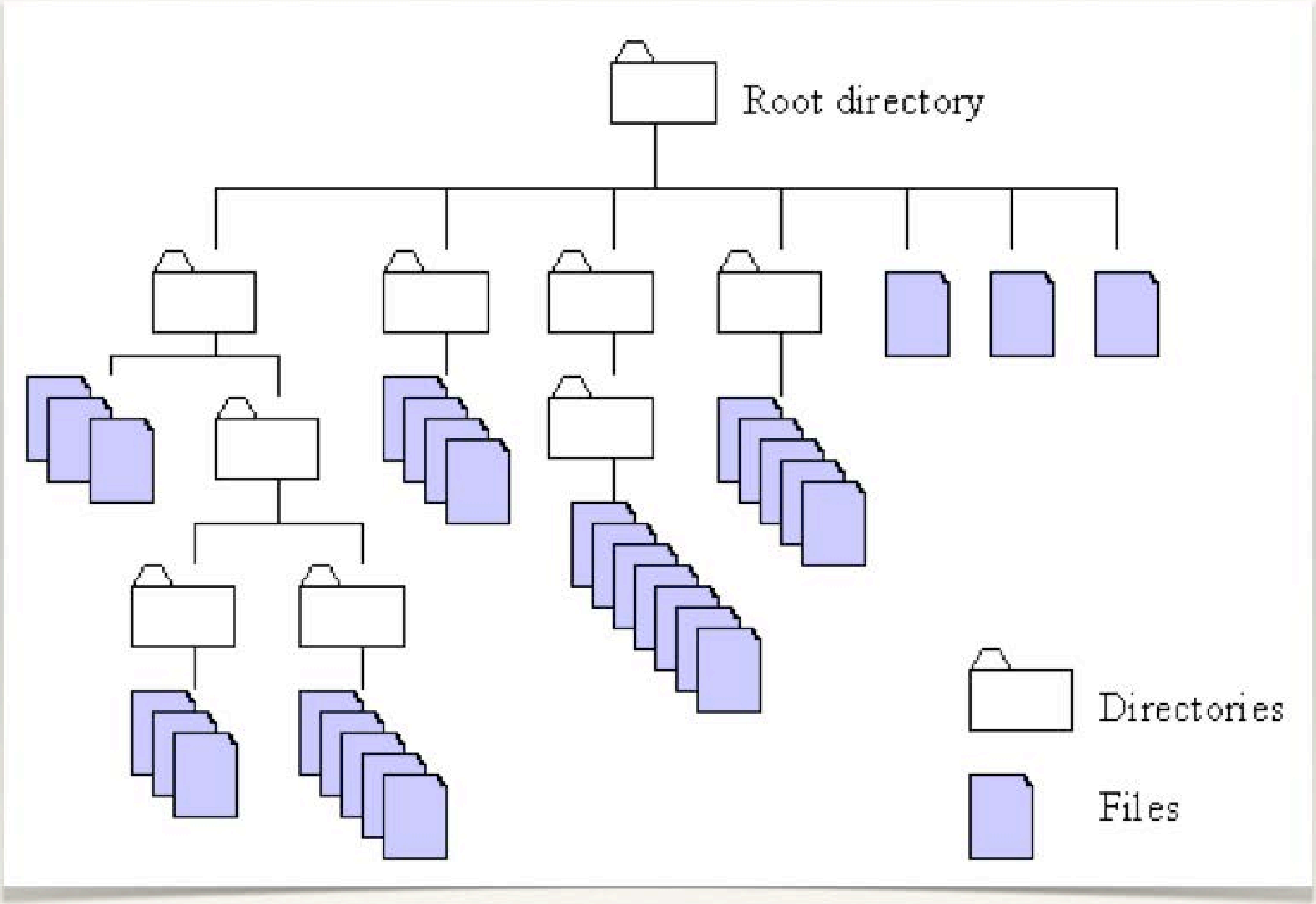
- List the files in a directory

Windows Console

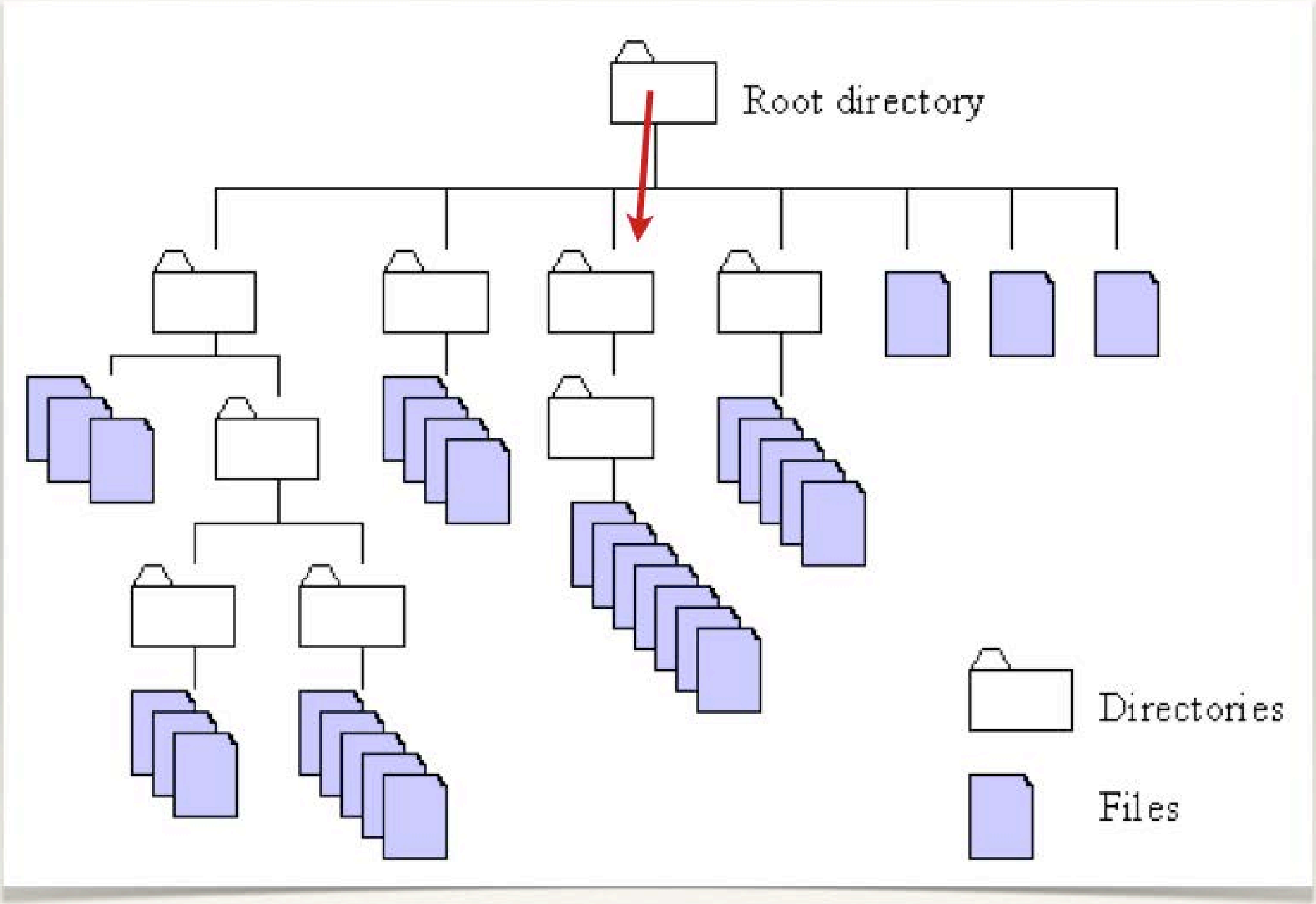


```
C:\Users\Franco\Documents>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents

04/08/2018  11:28 PM    <DIR>          .
04/08/2018  11:28 PM    <DIR>          ..
04/08/2018  11:28 PM           395,376 Anemone-Flower-5.jpg
04/08/2018  06:49 AM    <DIR>          Python Scripts
               1 File(s)        395,376 bytes
               3 Dir(s)  10,450,063,360 bytes free

C:\Users\Franco\Documents>_
```

The directories tree



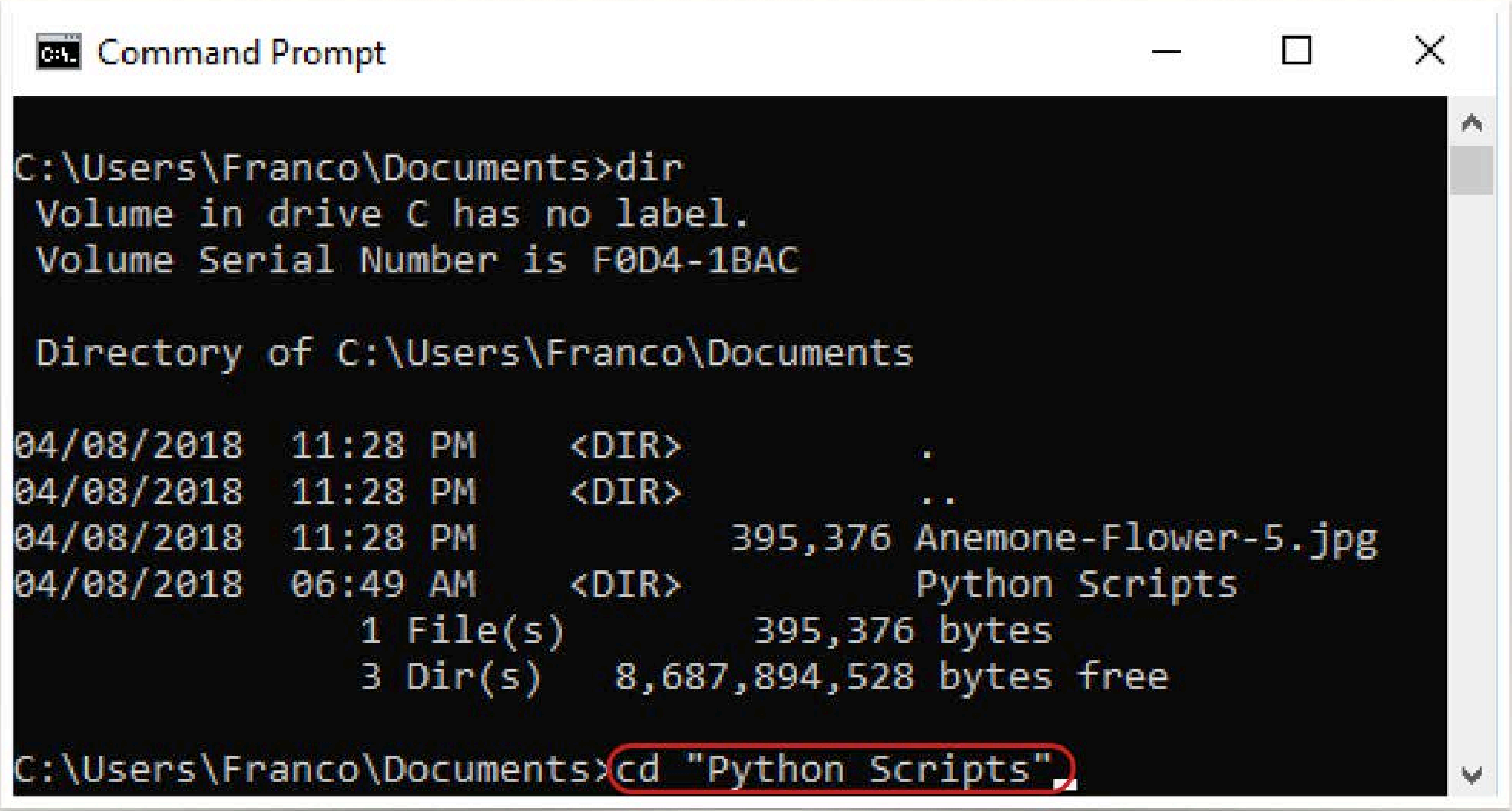
change directory



Windows Console

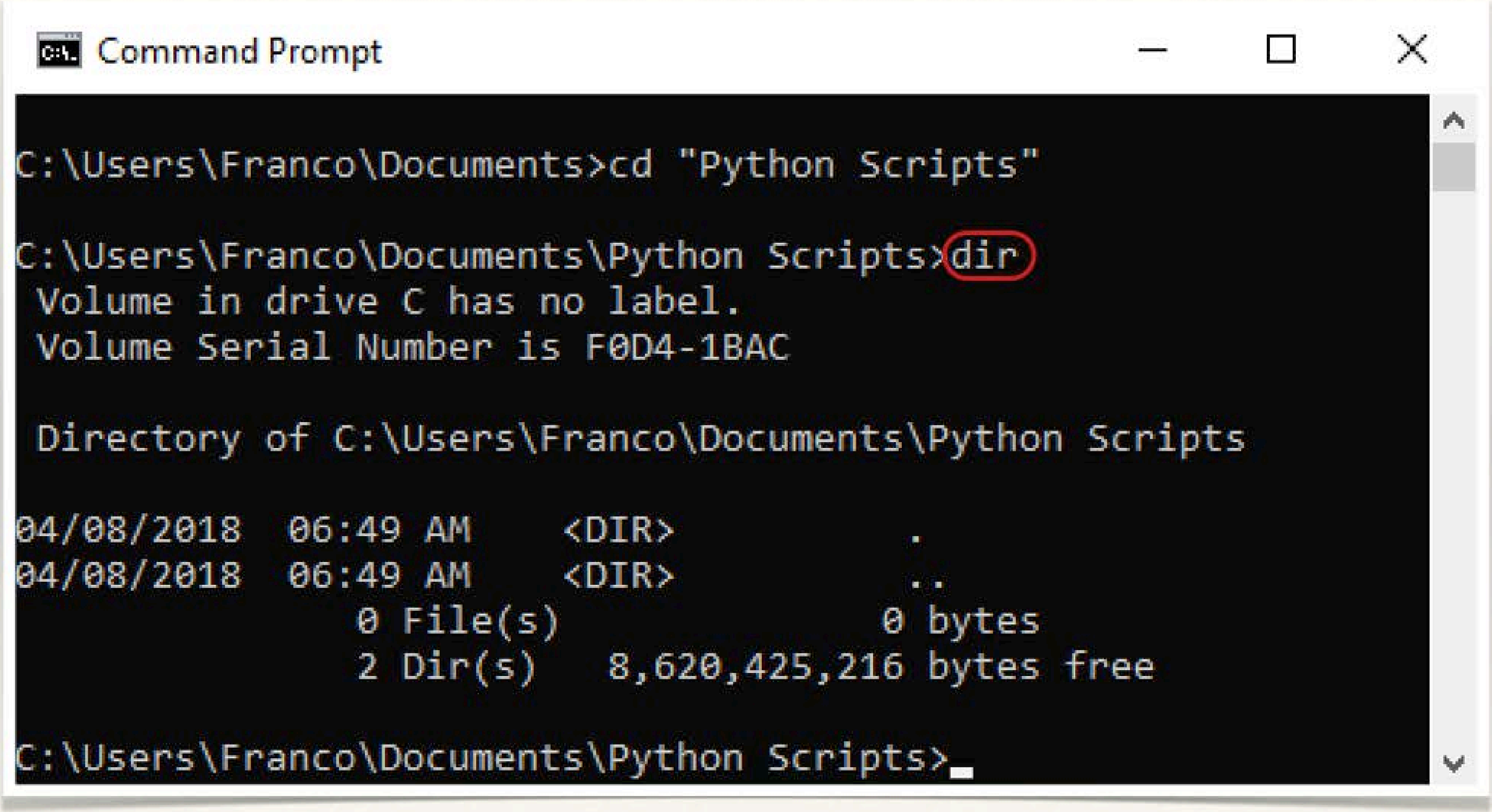
- change directory

Windows Console



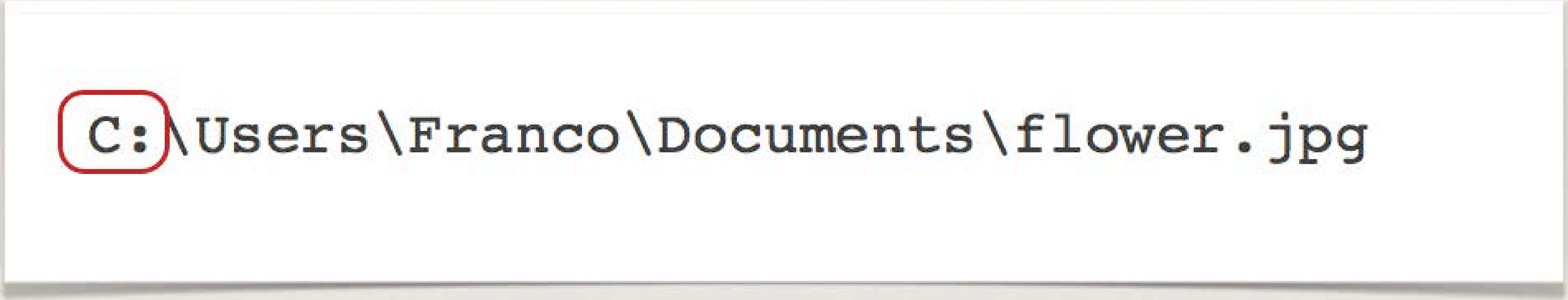
```
C:\Users\Franco\Documents>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents

04/08/2018  11:28 PM    <DIR>          .
04/08/2018  11:28 PM    <DIR>          ..
04/08/2018  11:28 PM           395,376 Anemone-Flower-5.jpg
04/08/2018  06:49 AM    <DIR>          Python Scripts
               1 File(s)        395,376 bytes
               3 Dir(s)   8,687,894,528 bytes free

C:\Users\Franco\Documents>cd "Python Scripts"
```

Windows Console



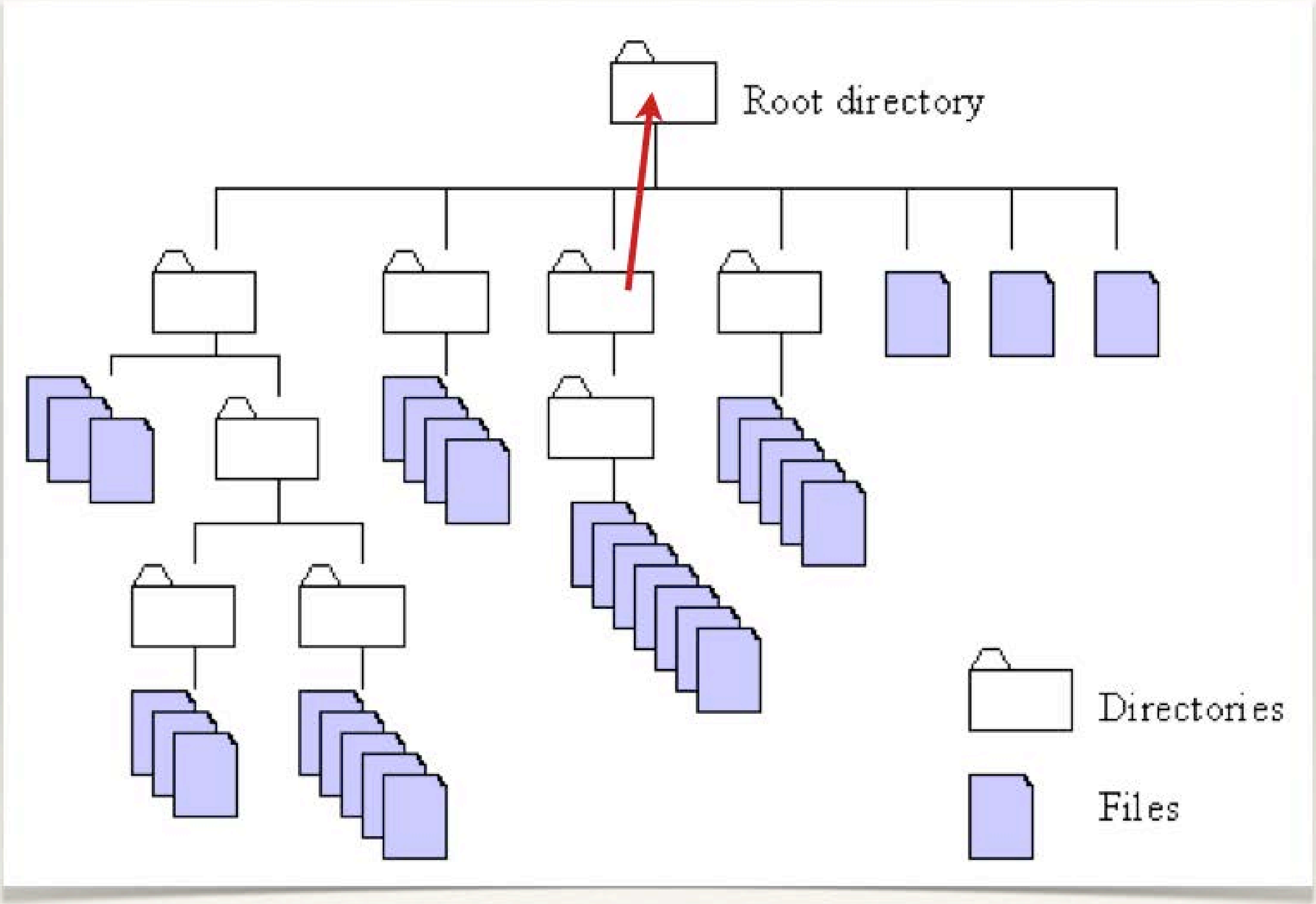
```
C:\Users\Franco\Documents>cd "Python Scripts"

C:\Users\Franco\Documents\Python Scripts>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents\Python Scripts

04/08/2018  06:49 AM    <DIR>          .
04/08/2018  06:49 AM    <DIR>          ..
               0 File(s)              0 bytes
               2 Dir(s)   8,620,425,216 bytes free

C:\Users\Franco\Documents\Python Scripts>
```

file path format (Windows)

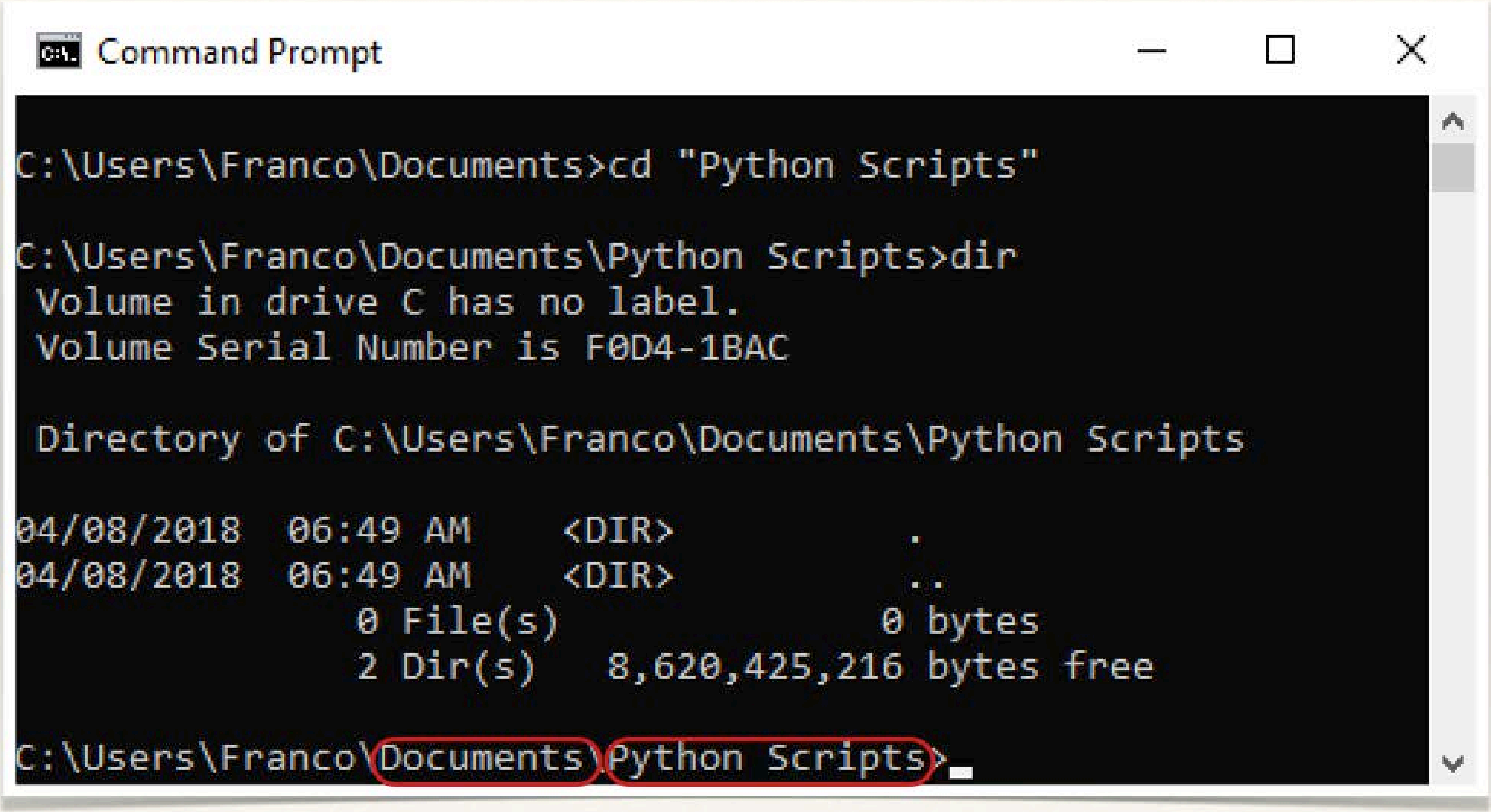


```
C:\Users\Franco\Documents\flower.jpg
```

change directory

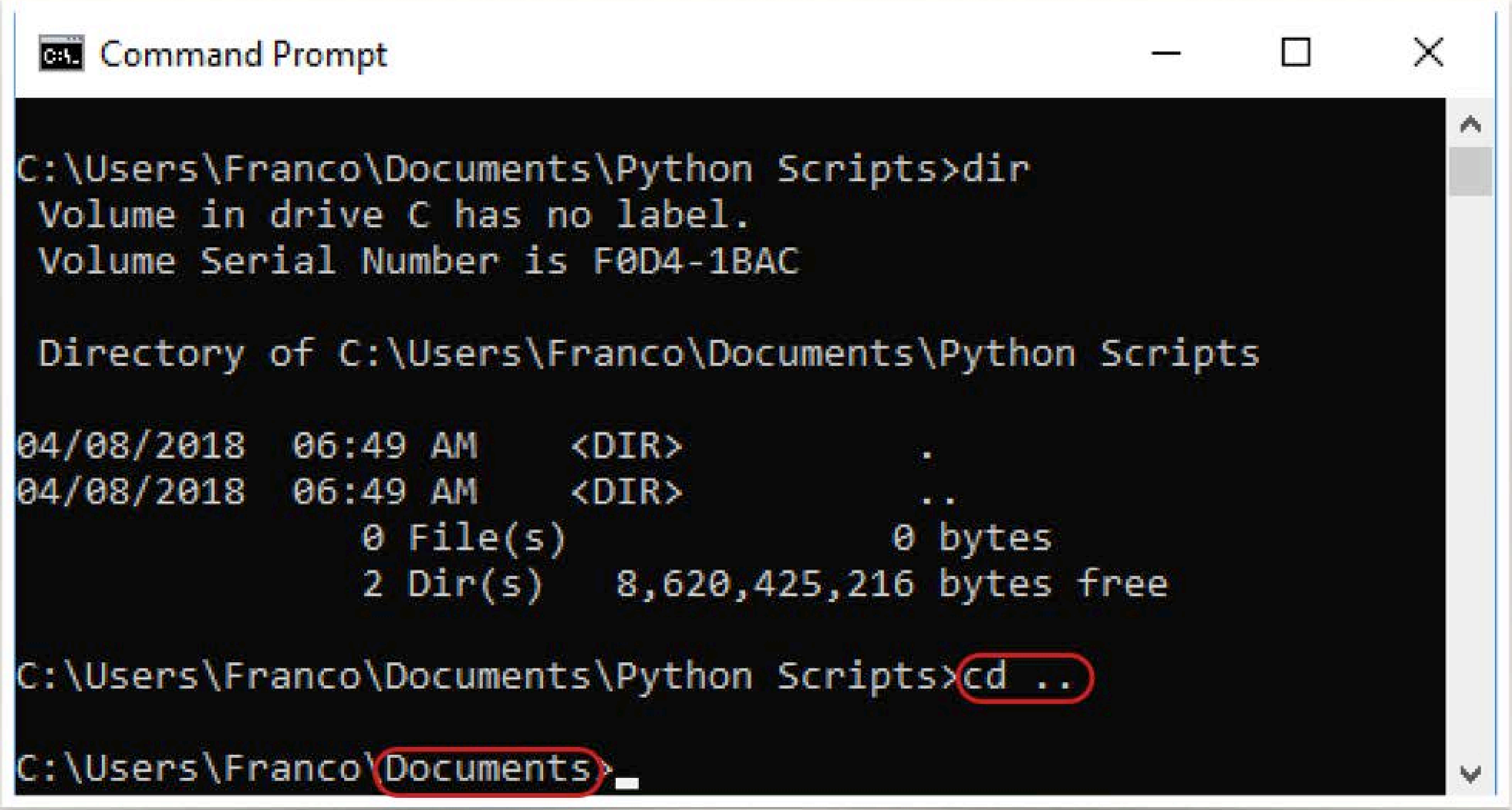


Windows Console



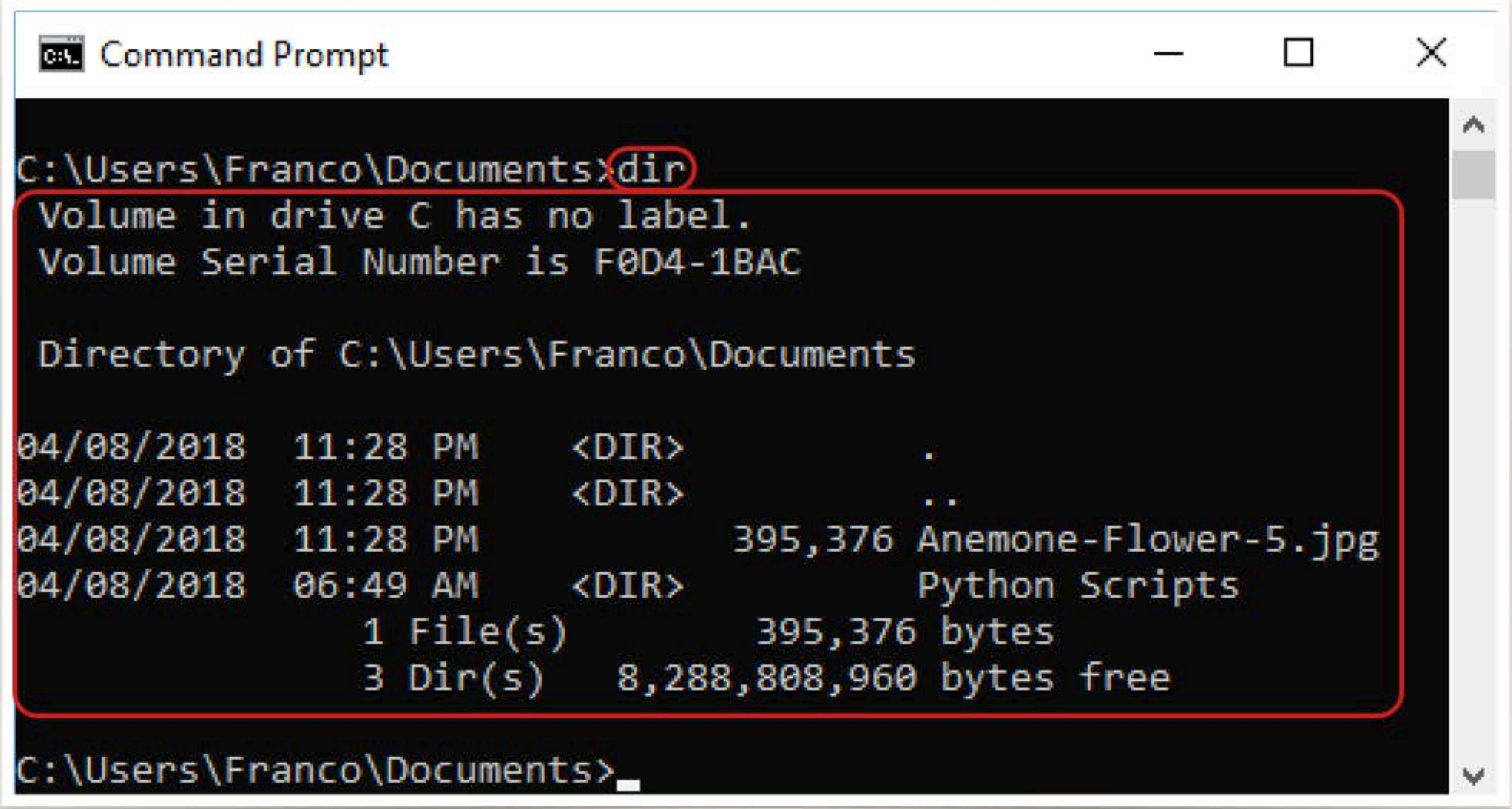
```
C:\Users\Franco\Documents>cd "Python Scripts"

C:\Users\Franco\Documents\Python Scripts>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents\Python Scripts

04/08/2018  06:49 AM    <DIR>          .
04/08/2018  06:49 AM    <DIR>          ..
               0 File(s)              0 bytes
               2 Dir(s)   8,620,425,216 bytes free

C:\Users\Franco\Documents\Python Scripts>_
```

Windows Console



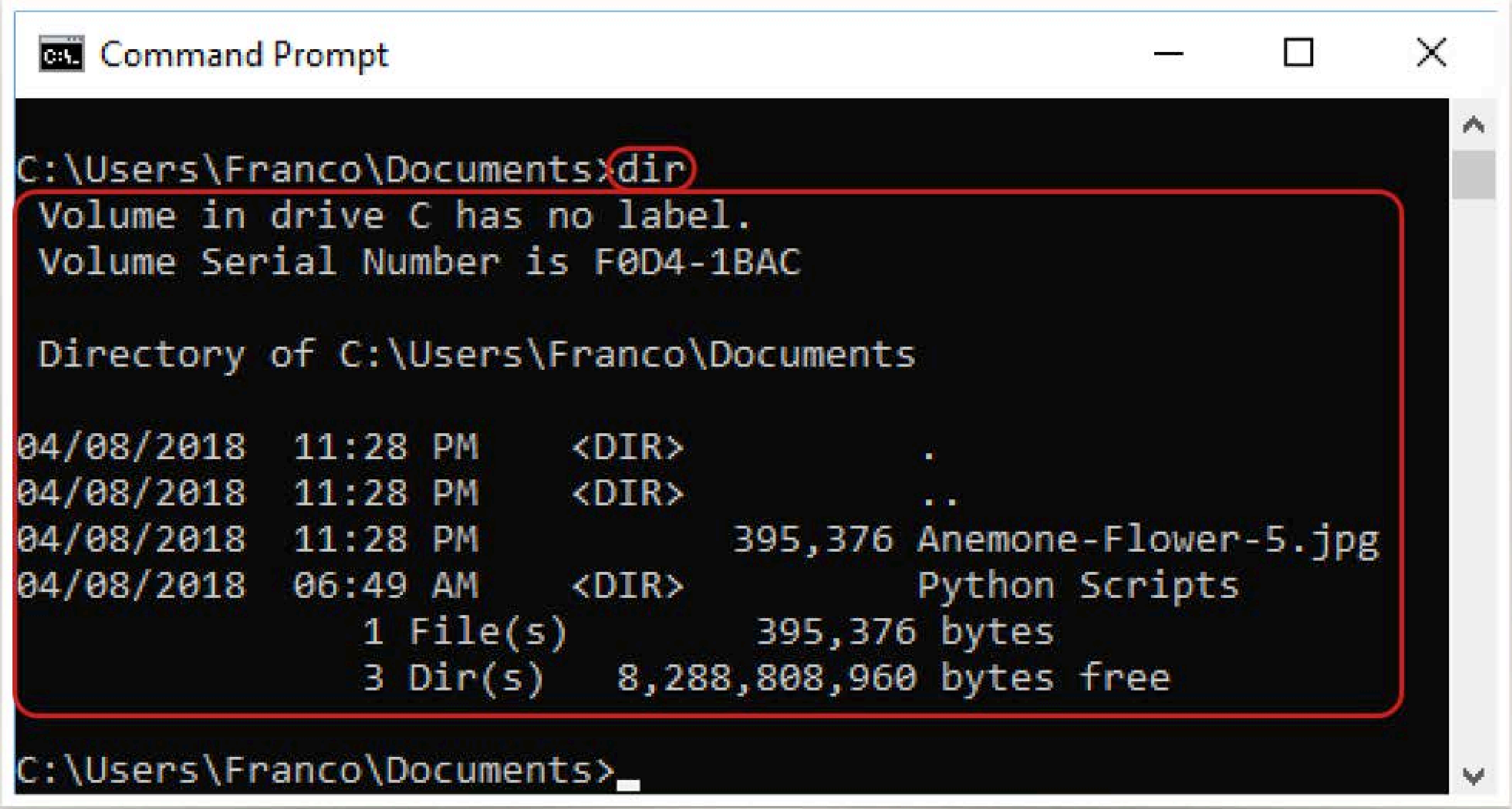
```
C:\Users\Franco\Documents\Python Scripts>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents\Python Scripts

04/08/2018  06:49 AM    <DIR>          .
04/08/2018  06:49 AM    <DIR>          ..
               0 File(s)              0 bytes
               2 Dir(s)   8,620,425,216 bytes free

C:\Users\Franco\Documents\Python Scripts>cd ..

C:\Users\Franco\Documents>_
```

Windows Console



```
C:\Users\Franco\Documents>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents

04/08/2018  11:28 PM    <DIR>          .
04/08/2018  11:28 PM    <DIR>          ..
04/08/2018  11:28 PM           395,376 Anemone-Flower-5.jpg
04/08/2018  06:49 AM    <DIR>          Python Scripts
               1 File(s)        395,376 bytes
               3 Dir(s)   8,288,808,960 bytes free

C:\Users\Franco\Documents>_
```

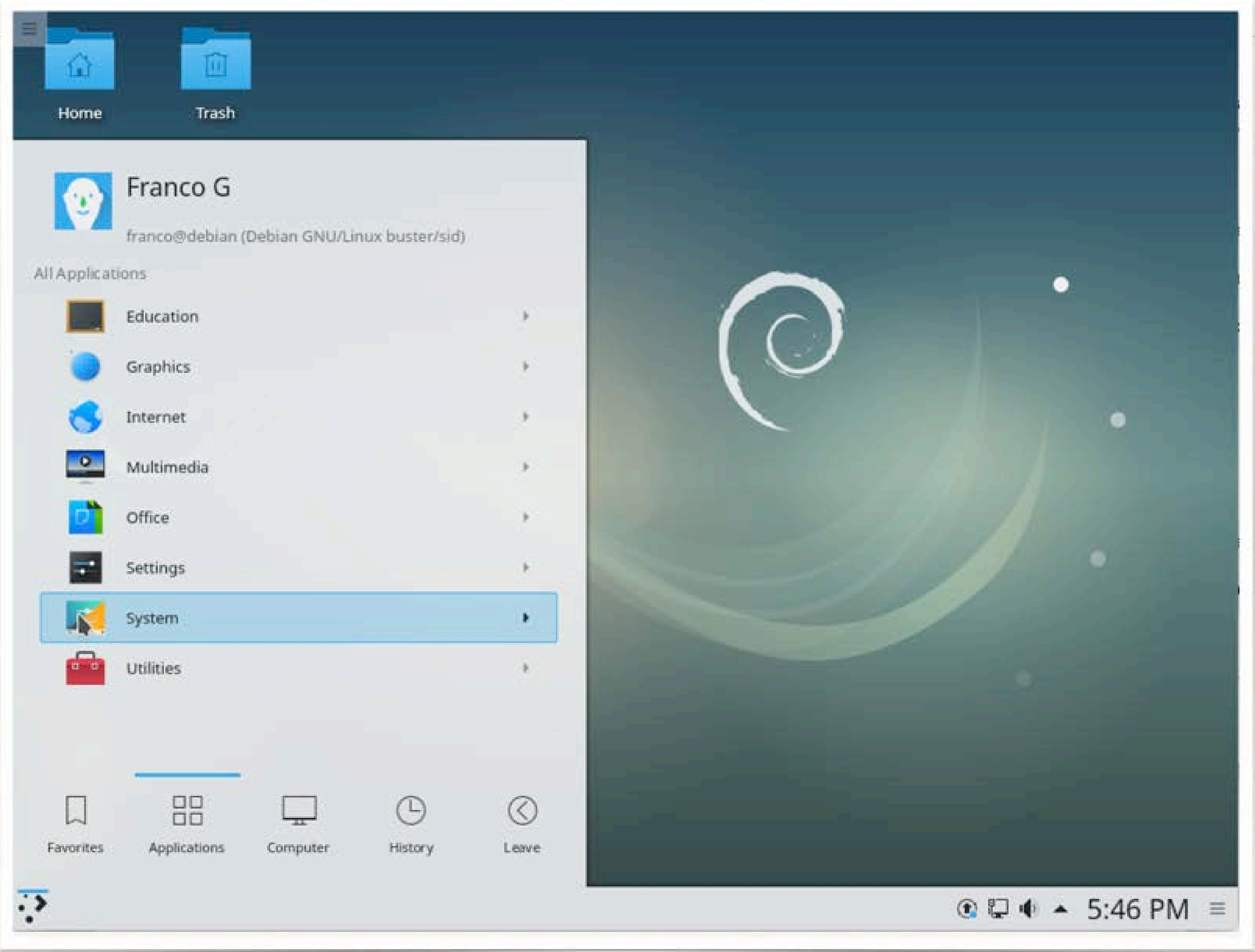
Windows Console



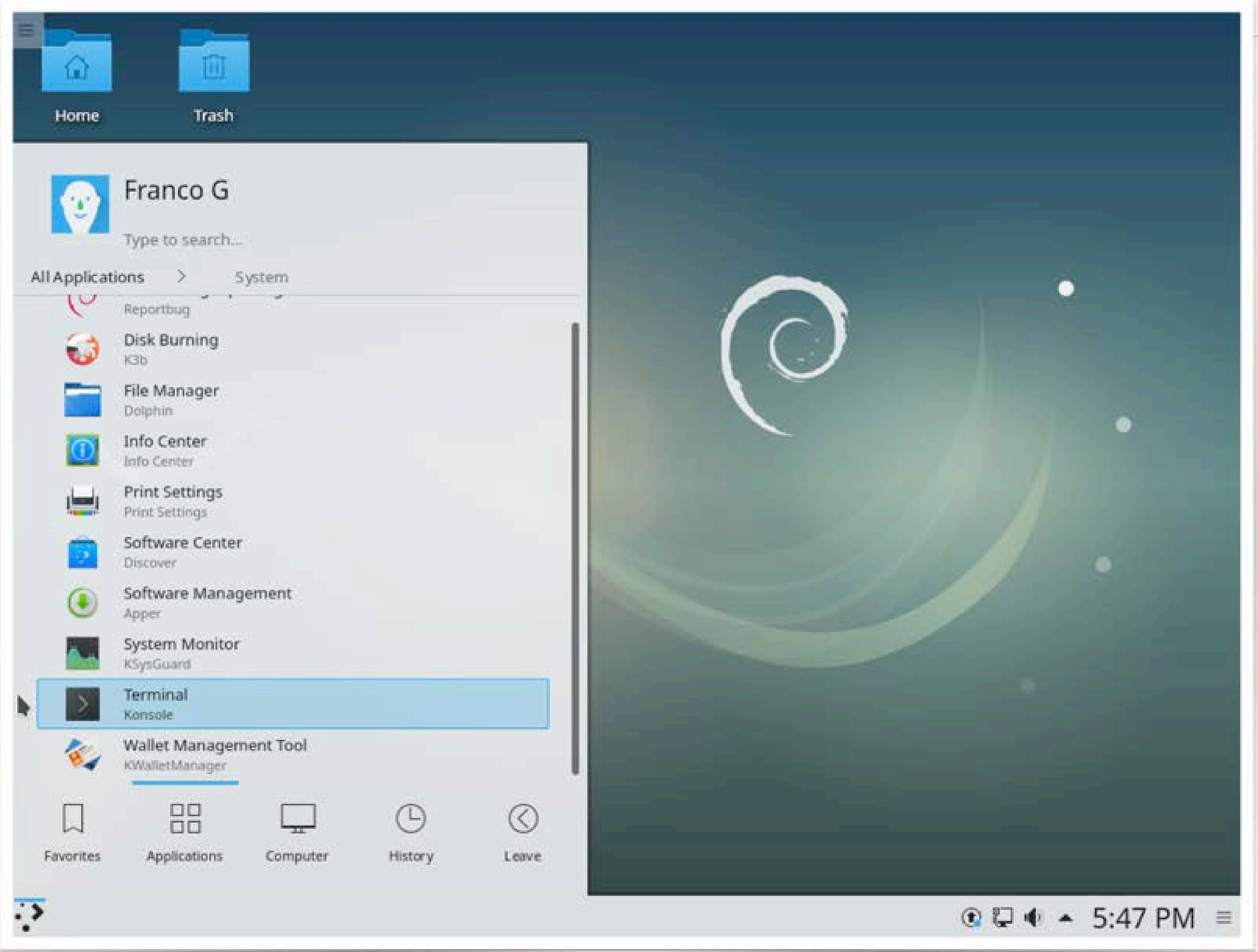
```
C:\Users\Franco\Documents>dir
 Volume in drive C has no label.
 Volume Serial Number is F0D4-1BAC

 Directory of C:\Users\Franco\Documents

04/08/2018  11:28 PM    <DIR>          .
04/08/2018  11:28 PM    <DIR>          ..
04/08/2018  11:28 PM           395,376 Anemone-Flower-5.jpg
04/08/2018  06:49 AM    <DIR>          Python Scripts
               1 File(s)        395,376 bytes
               3 Dir(s)   8,288,808,960 bytes free

C:\Users\Franco\Documents>_
```

linux

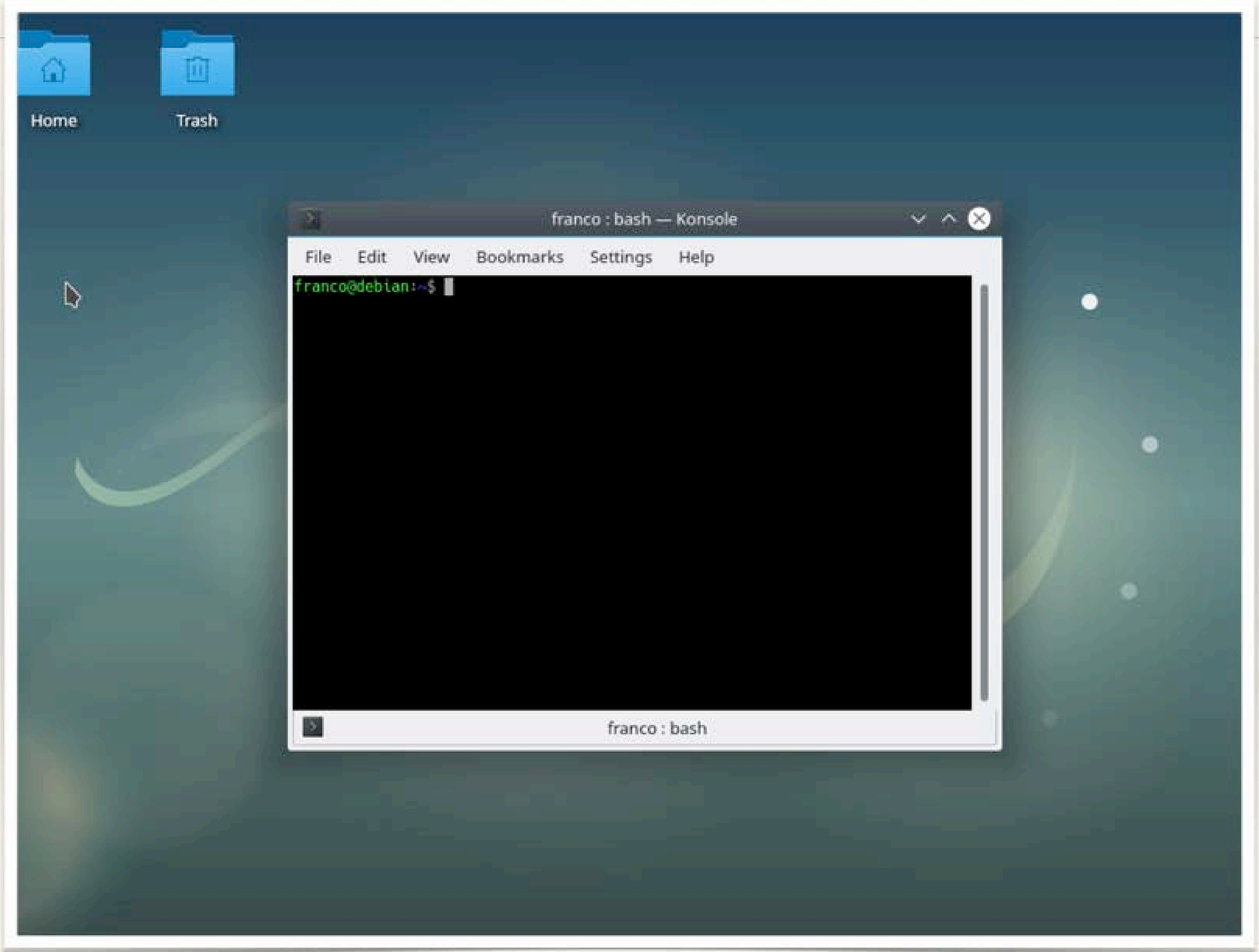
Debian with KDE



Debian with KDE

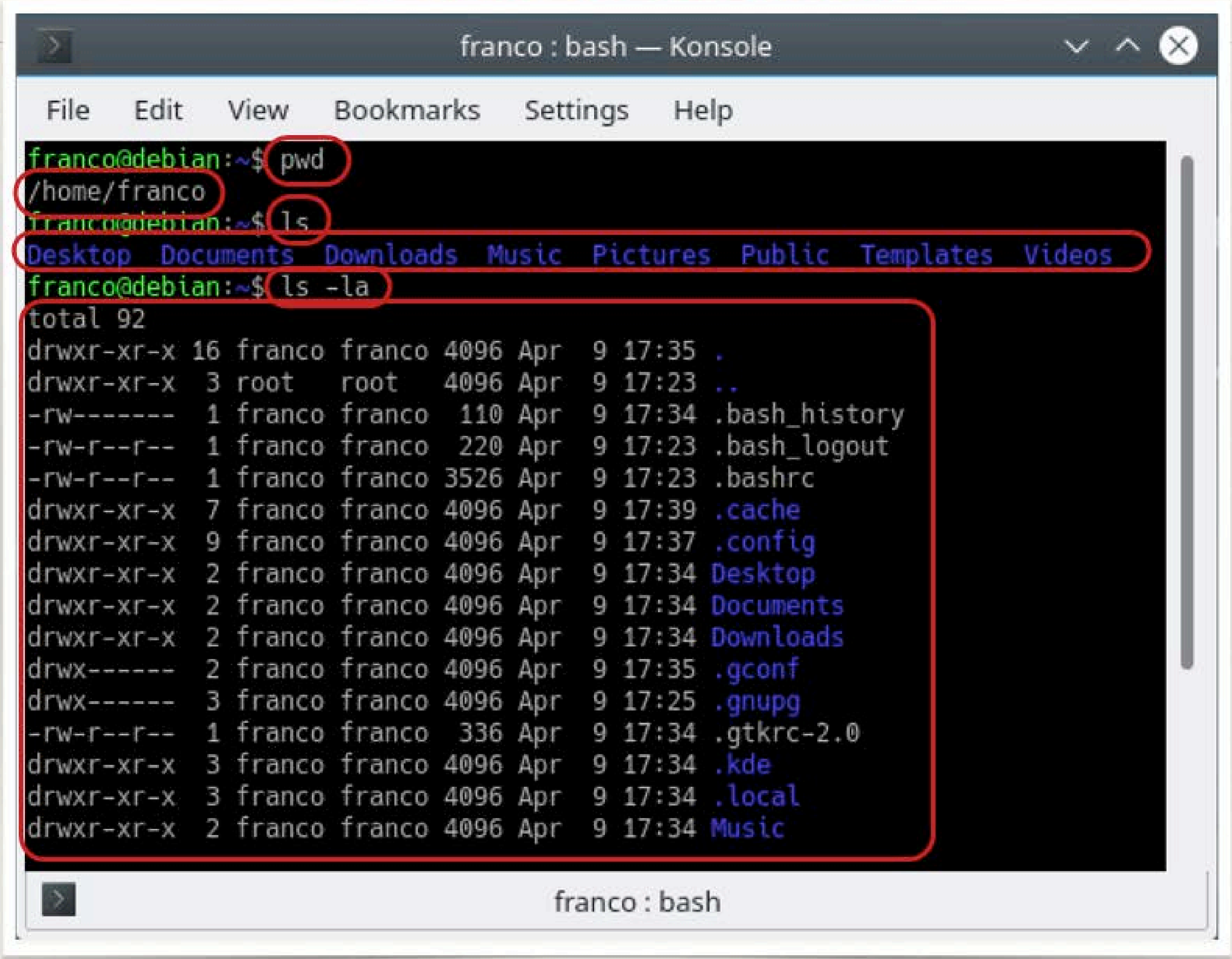


Debian with KDE



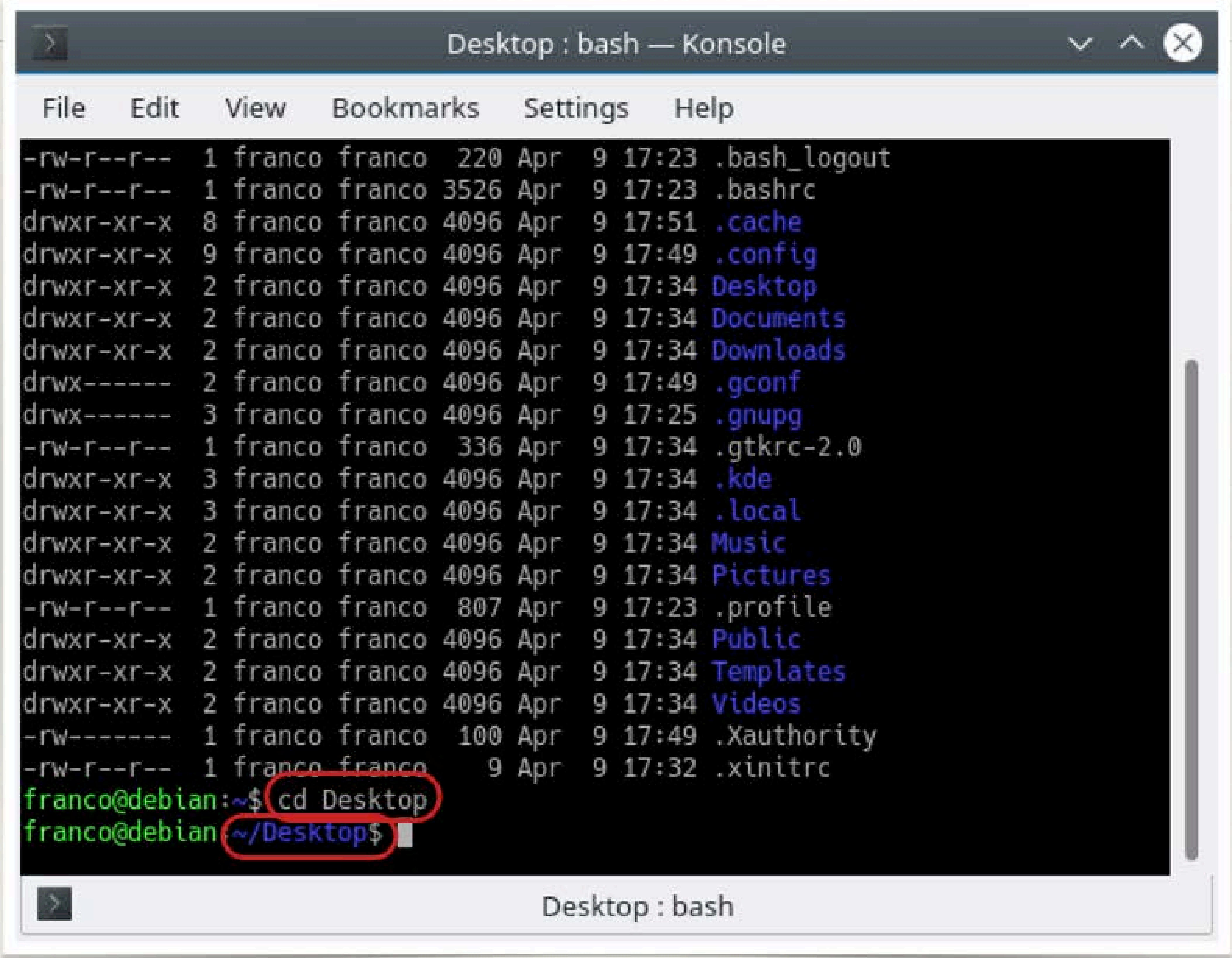
```
franco@debian:~$
```

linux Console



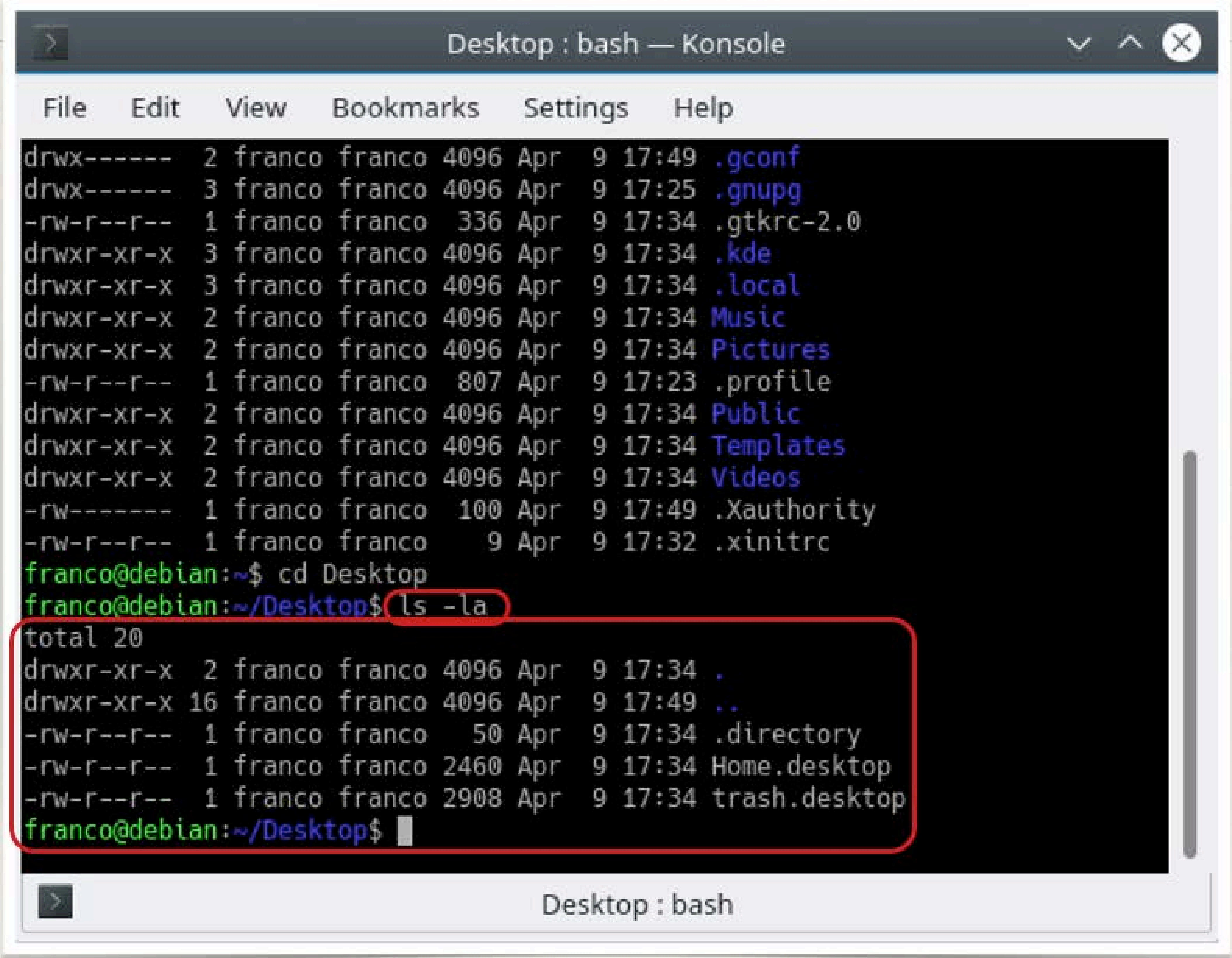
```
franco@debian:~$ pwd
/home/franco
franco@debian:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
franco@debian:~$ ls -la
total 92
drwxr-xr-x 16 franco franco 4096 Apr  9 17:35 .
drwxr-xr-x  3 root   root   4096 Apr  9 17:23 ..
-rw-------  1 franco franco  110 Apr  9 17:34 .bash_history
-rw-r--r--  1 franco franco  220 Apr  9 17:23 .bash_logout
-rw-r--r--  1 franco franco 3526 Apr  9 17:23 .bashrc
drwxr-xr-x  7 franco franco 4096 Apr  9 17:39 .cache
drwxr-xr-x  9 franco franco 4096 Apr  9 17:37 .config
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Desktop
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Documents
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Downloads
drwx------  2 franco franco 4096 Apr  9 17:35 .gconf
drwx------  3 franco franco 4096 Apr  9 17:25 .gnupg
-rw-r--r--  1 franco franco  336 Apr  9 17:34 .gtkrc-2.0
drwxr-xr-x  3 franco franco 4096 Apr  9 17:34 .kde
drwxr-xr-x  3 franco franco 4096 Apr  9 17:34 .local
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Music
```

linux Console



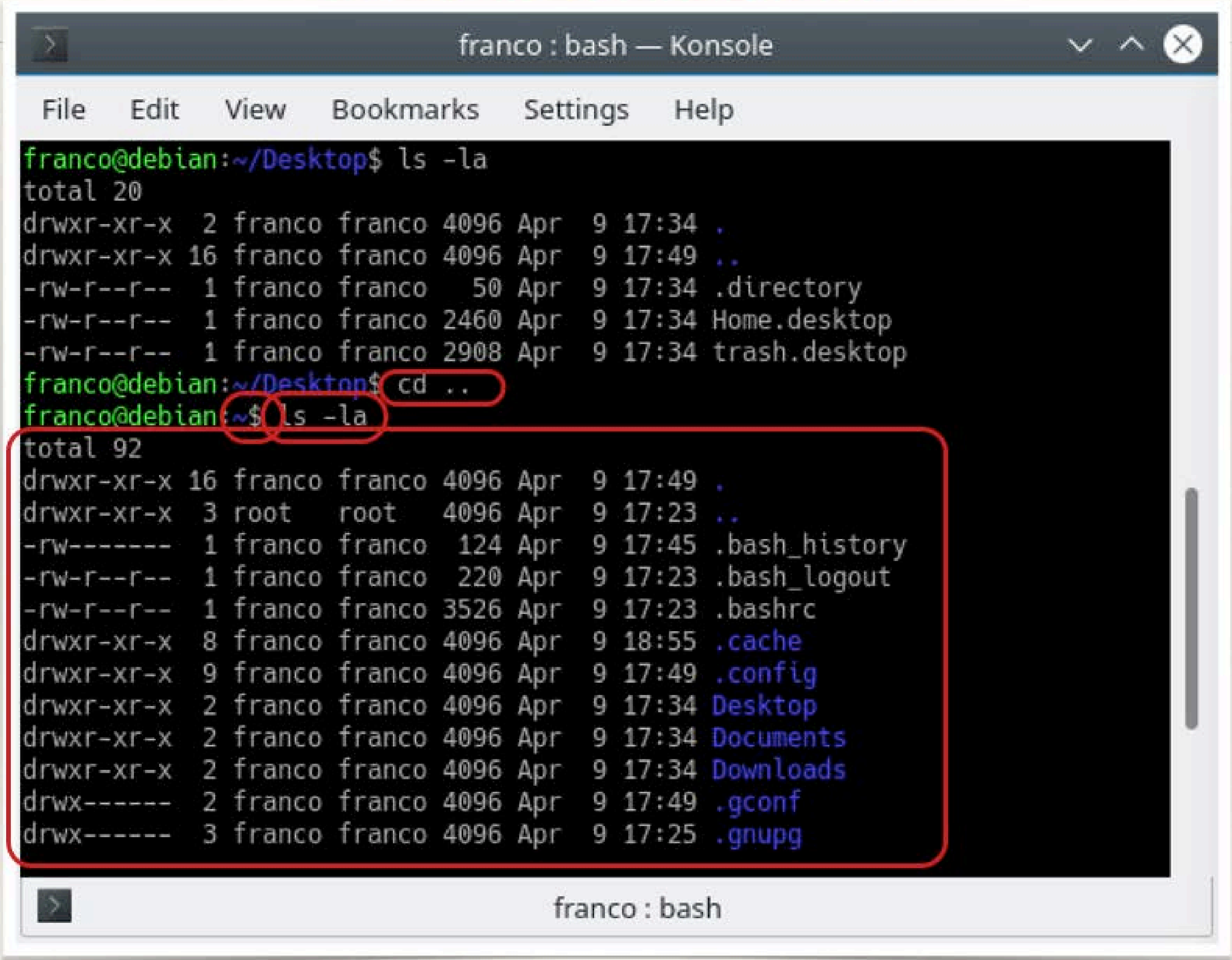
```
-rw-r--r--  1 franco franco  220 Apr  9 17:23 .bash_logout
-rw-r--r--  1 franco franco 3526 Apr  9 17:23 .bashrc
drwxr-xr-x  8 franco franco 4096 Apr  9 17:51 .cache
drwxr-xr-x  9 franco franco 4096 Apr  9 17:49 .config
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Desktop
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Documents
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Downloads
drwx------  2 franco franco 4096 Apr  9 17:49 .gconf
drwx------  3 franco franco 4096 Apr  9 17:25 .gnupg
-rw-r--r--  1 franco franco  336 Apr  9 17:34 .gtkrc-2.0
drwxr-xr-x  3 franco franco 4096 Apr  9 17:34 .kde
drwxr-xr-x  3 franco franco 4096 Apr  9 17:34 .local
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Music
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Pictures
-rw-r--r--  1 franco franco  807 Apr  9 17:23 .profile
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Public
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Templates
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Videos
-rw-------  1 franco franco  100 Apr  9 17:49 .Xauthority
-rw-r--r--  1 franco franco    9 Apr  9 17:32 .xinitrc
franco@debian:~$ cd Desktop
franco@debian:~/Desktop$ 
```

linux Console



```
drwx------  2 franco franco 4096 Apr  9 17:49 .gconf
drwx------  3 franco franco 4096 Apr  9 17:25 .gnupg
-rw-r--r--  1 franco franco  336 Apr  9 17:34 .gtkrc-2.0
drwxr-xr-x  3 franco franco 4096 Apr  9 17:34 .kde
drwxr-xr-x  3 franco franco 4096 Apr  9 17:34 .local
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Music
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Pictures
-rw-r--r--  1 franco franco  807 Apr  9 17:23 .profile
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Public
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Templates
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Videos
-rw-------  1 franco franco  100 Apr  9 17:49 .Xauthority
-rw-r--r--  1 franco franco    9 Apr  9 17:32 .xinitrc
franco@debian:~$ cd Desktop
franco@debian:~/Desktop$ ls -la
total 20
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 .
drwxr-xr-x 16 franco franco 4096 Apr  9 17:49 ..
-rw-r--r--  1 franco franco   50 Apr  9 17:34 .directory
-rw-r--r--  1 franco franco 2460 Apr  9 17:34 Home.desktop
-rw-r--r--  1 franco franco 2908 Apr  9 17:34 trash.desktop
franco@debian:~/Desktop$ 
```

linux Console



```
franco@debian:~/Desktop$ ls -la
total 20
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 .
drwxr-xr-x 16 franco franco 4096 Apr  9 17:49 ..
-rw-r--r--  1 franco franco   50 Apr  9 17:34 .directory
-rw-r--r--  1 franco franco 2460 Apr  9 17:34 Home.desktop
-rw-r--r--  1 franco franco 2908 Apr  9 17:34 trash.desktop
franco@debian:~/Desktop$ cd ..
franco@debian:~$ ls -la
total 92
drwxr-xr-x 16 franco franco 4096 Apr  9 17:49 .
drwxr-xr-x  3 root   root   4096 Apr  9 17:23 ..
-rw-------  1 franco franco  124 Apr  9 17:45 .bash_history
-rw-r--r--  1 franco franco  220 Apr  9 17:23 .bash_logout
-rw-r--r--  1 franco franco 3526 Apr  9 17:23 .bashrc
drwxr-xr-x  8 franco franco 4096 Apr  9 18:55 .cache
drwxr-xr-x  9 franco franco 4096 Apr  9 17:49 .config
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Desktop
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Documents
drwxr-xr-x  2 franco franco 4096 Apr  9 17:34 Downloads
drwx------  2 franco franco 4096 Apr  9 17:49 .gconf
drwx------  3 franco franco 4096 Apr  9 17:25 .gnupg
```

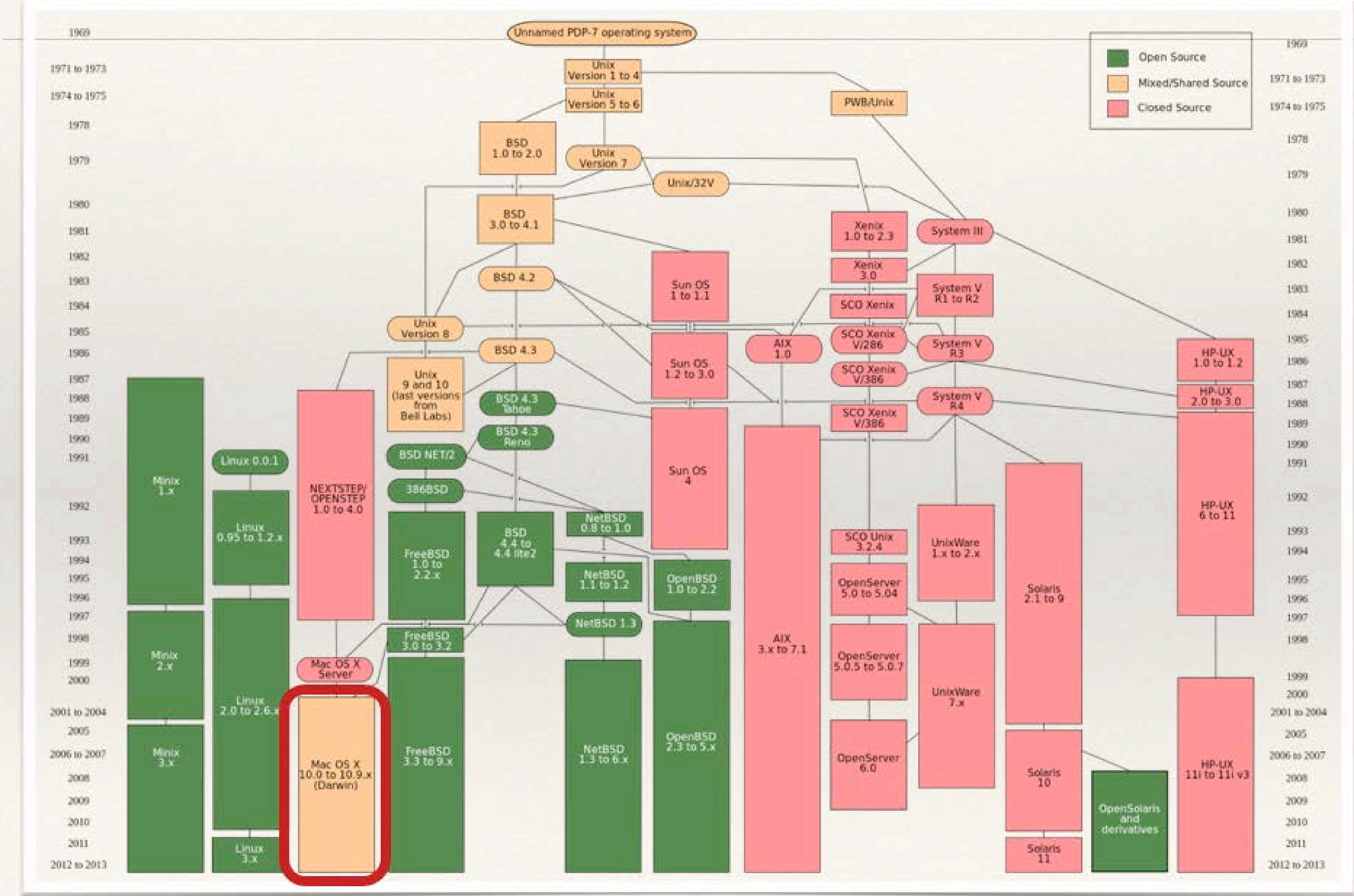
file path format (Unix/Linux)

```
/Users/Franco/Documents/flower.jpg
```

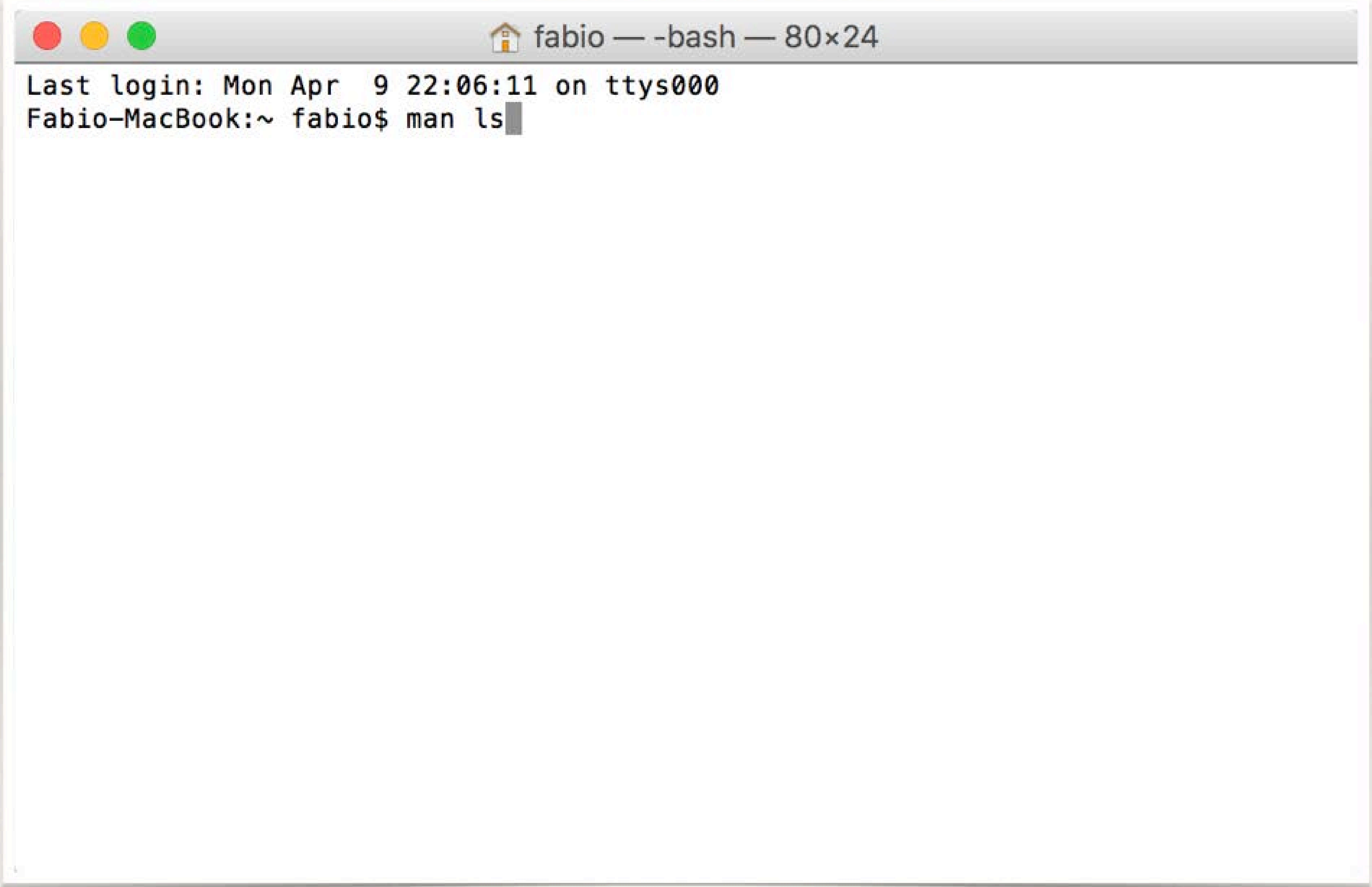
```
~/Documents/flower.jpg
```

```
./flower.jpg
```

MacOS

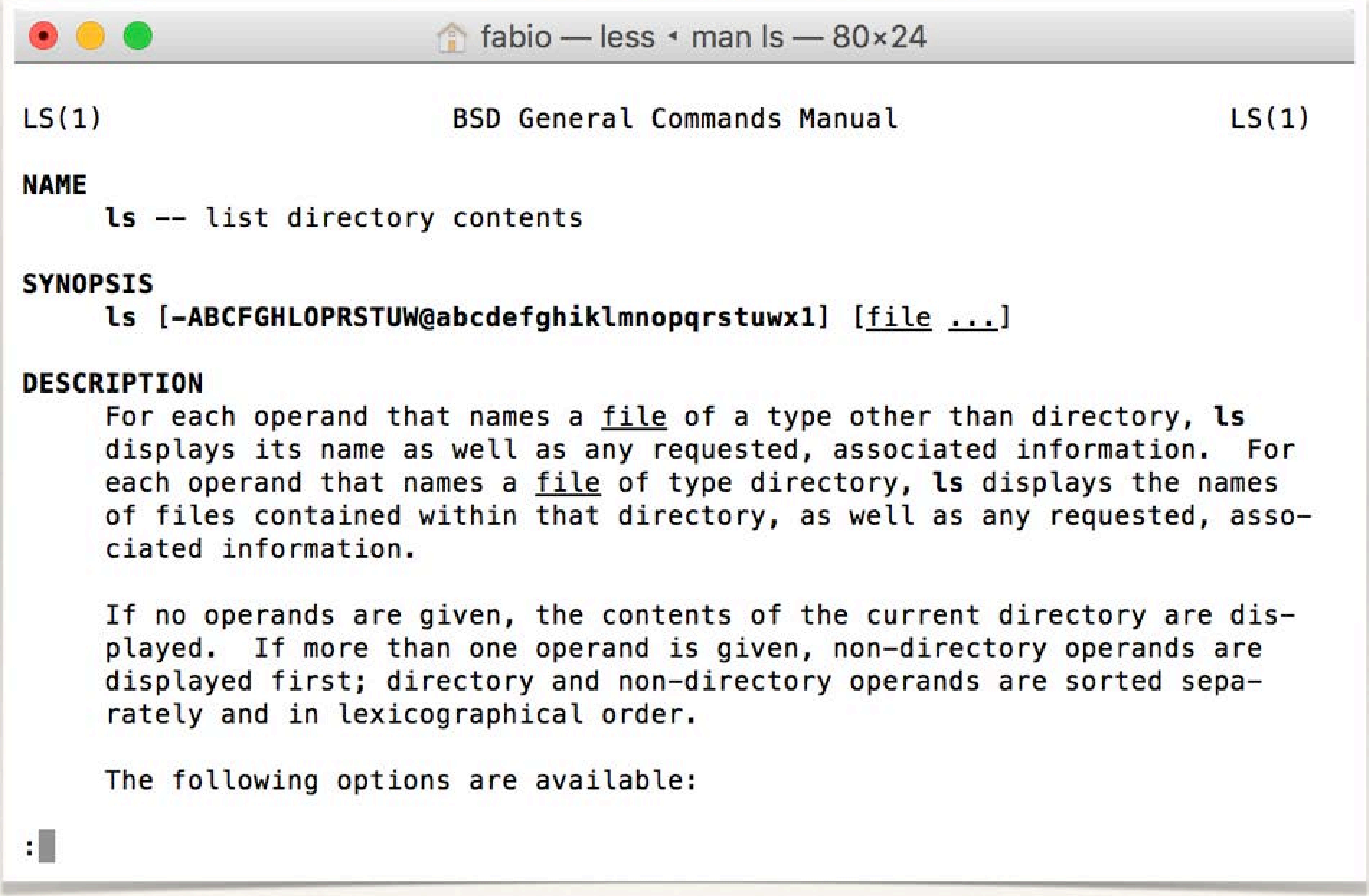

Unnamed PDP-7 operating system
Open Source
Mixed/Shared Source
Closed Source
1969
1971 to 1973
1974 to 1975
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001 to 2004
2005
2006 to 2007
2008
2009
2010
2011
2012 to 2013
Unix Version 1 to 4
Unix Version 5 to 6
PWB/Unix
BSD 1.0 to 2.0
Unix Version 7
Unix/32V
BSD 3.0 to 4.1
Xenix 1.0 to 2.3
System III
Xenix 3.0
Sun OS 1 to 1.1
BSD 4.2
System V R1 to R2
SCO Xenix
Unix Version 8
SCO Xenix V/286
System V R3
AIX 1.0
BSD 4.3
Sun OS 1.2 to 3.0
HP-UX 1.0 to 1.2
SCO Xenix V/386
Unix 9 and 10 (last versions from Bell Labs)
BSD 4.3 Tahoe
System V R4
HP-UX 2.0 to 3.0
SCO Xenix V/386
BSD 4.3 Reno
Minix 1.x
Linux 0.0.1
NEXTSTEP/ OPENSTEP 1.0 to 4.0
BSD NET/2
Sun OS 4
386BSD
HP-UX 6 to 11
NetBSD 0.8 to 1.0
Linux 0.95 to 1.2.x
FreeBSD 1.0 to 2.2.x
BSD 4.4 to 4.4 lite2
SCO Unix 3.2.4
UnixWare 1.x to 2.x
NetBSD 1.1 to 1.2
OpenBSD 1.0 to 2.2
OpenServer 5.0 to 5.04
Solaris 2.1 to 9
NetBSD 1.3
Minix 2.x
FreeBSD 3.0 to 3.2
AIX 3.x to 7.1
OpenServer 5.0.5 to 5.0.7
Mac OS X Server
UnixWare 7.x
Linux 2.0 to 2.6.x
Minix 3.x
Mac OS X 10.0 to 10.9.x (Darwin)
FreeBSD 3.3 to 9.x
NetBSD 1.3 to 6.x
OpenBSD 2.3 to 5.x
OpenServer 6.0
Solaris 10
HP-UX 11i to 11i v3
OpenSolaris and derivatives
Linux 3.x
Solaris 11

MacOS Console



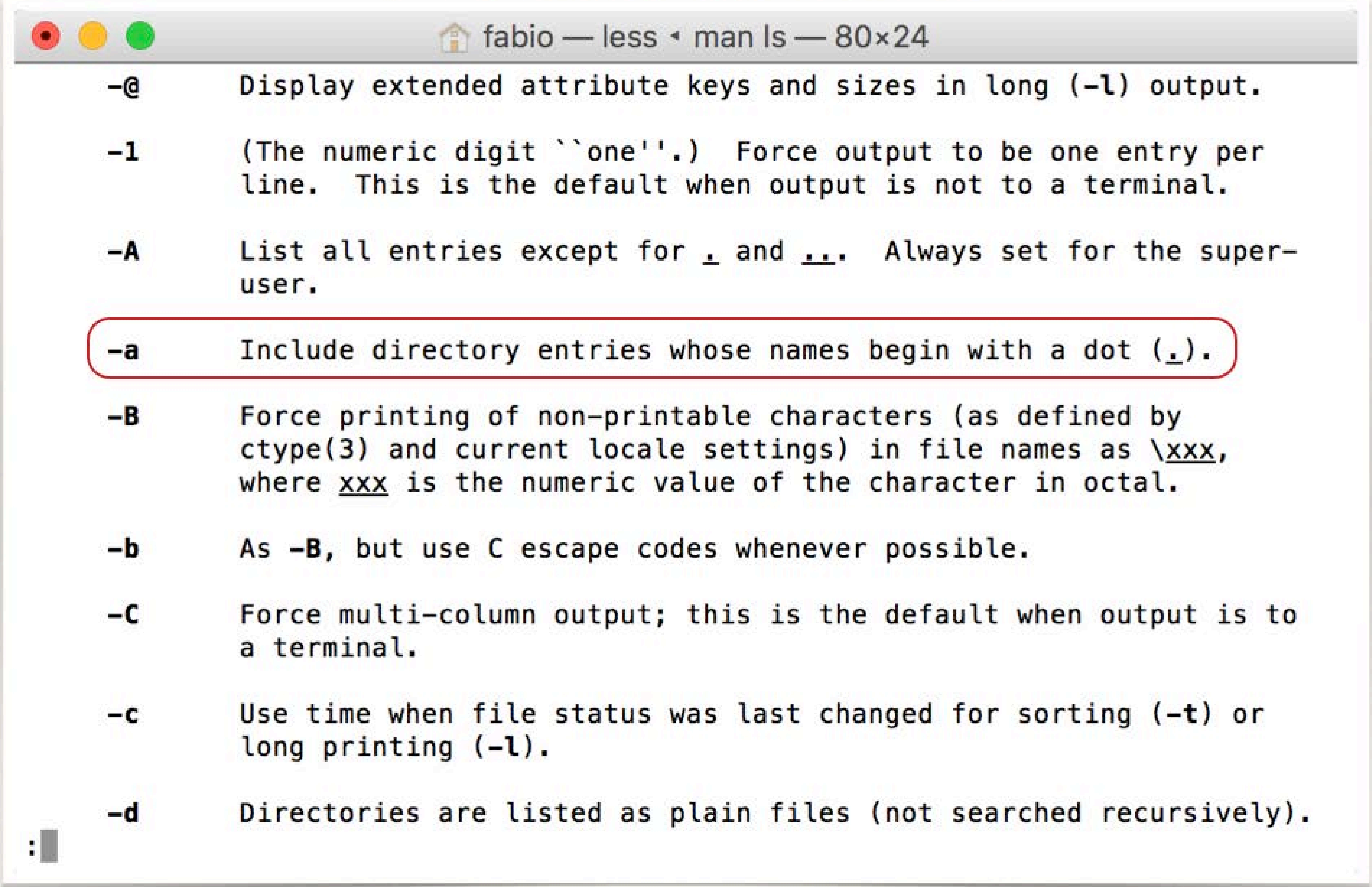
```
Last login: Mon Apr  9 22:06:11 on ttys000
Fabio-MacBook:~ fabio$ man ls
```

MacOS Console



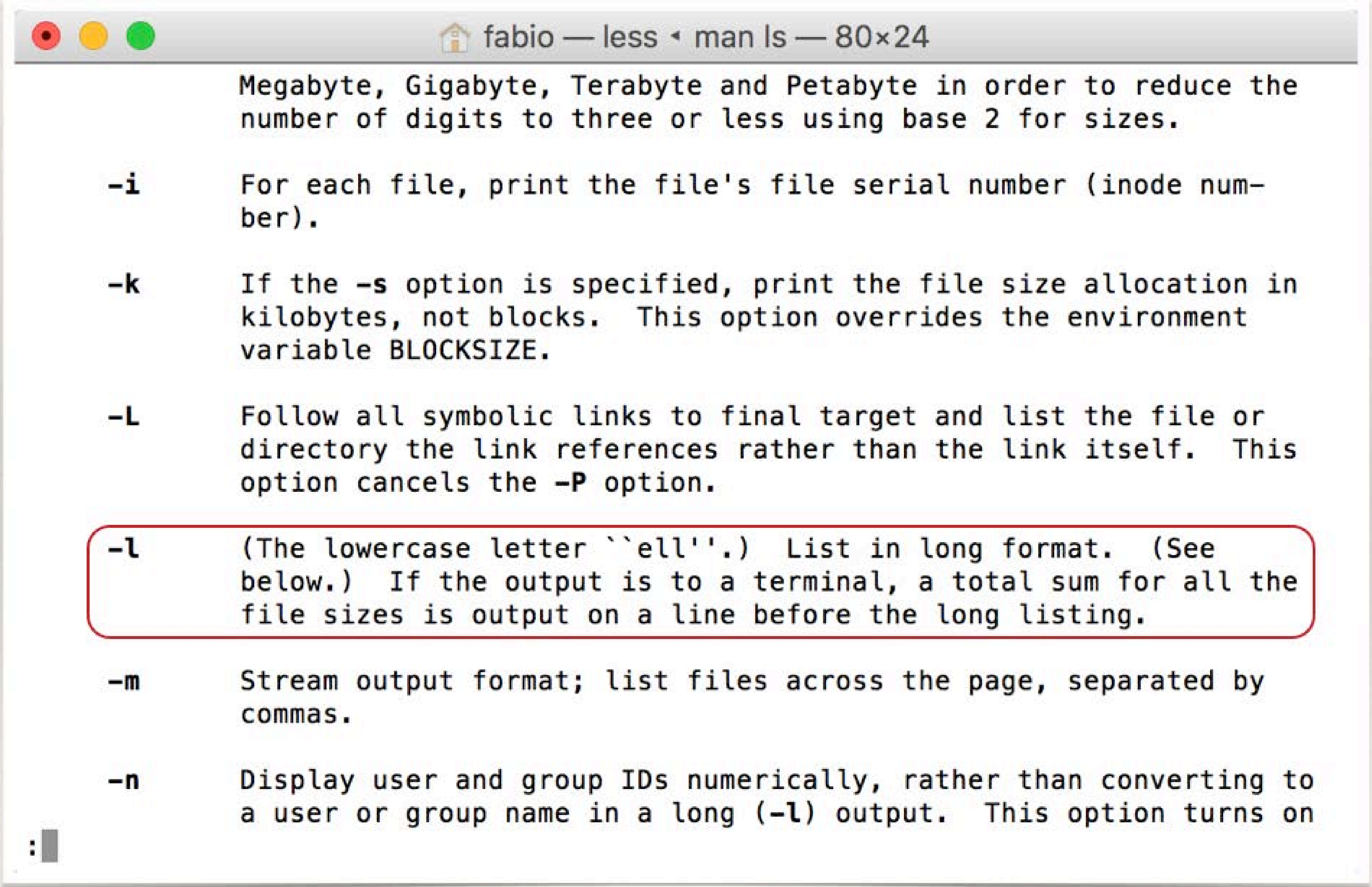
```
LS(1)                    BSD General Commands Manual                   LS(1)

NAME
     ls -- list directory contents

SYNOPSIS
     ls [-ABCFGHLOPRSTUW@abcdefghiklmnopqrstuwx1] [file ...]

DESCRIPTION
     For each operand that names a file of a type other than directory, ls
     displays its name as well as any requested, associated information.  For
     each operand that names a file of type directory, ls displays the names
     of files contained within that directory, as well as any requested, asso-
     ciated information.

     If no operands are given, the contents of the current directory are dis-
     played.  If more than one operand is given, non-directory operands are
     displayed first; directory and non-directory operands are sorted sepa-
     rately and in lexicographical order.

     The following options are available:

:
```

MacOS Console



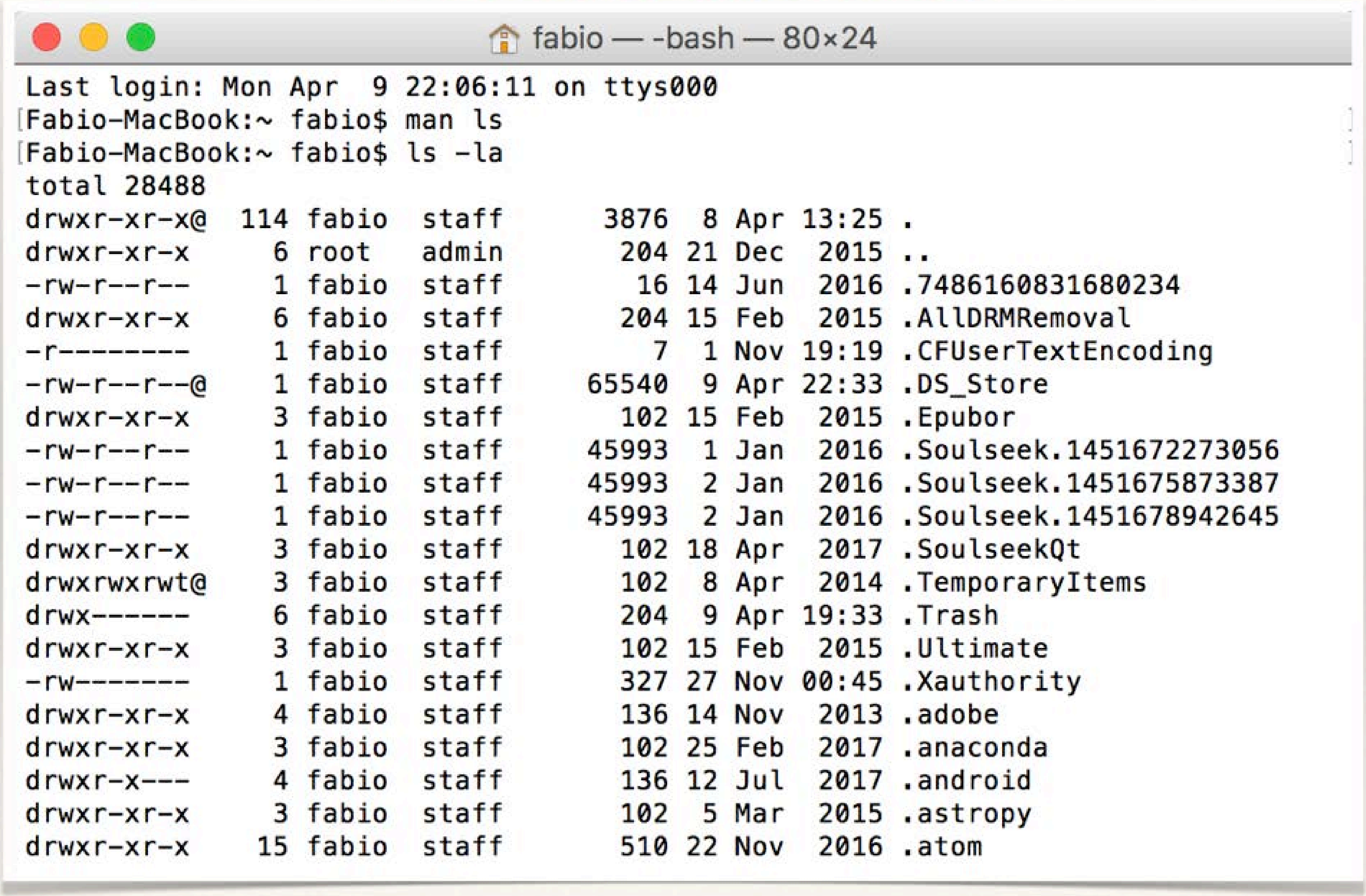
```
     -@      Display extended attribute keys and sizes in long (-l) output.

     -1      (The numeric digit ``one''.)  Force output to be one entry per
             line.  This is the default when output is not to a terminal.

     -A      List all entries except for . and ...  Always set for the super-
             user.

     -a      Include directory entries whose names begin with a dot (.).

     -B      Force printing of non-printable characters (as defined by
             ctype(3) and current locale settings) in file names as \xxx,
             where xxx is the numeric value of the character in octal.

     -b      As -B, but use C escape codes whenever possible.

     -C      Force multi-column output; this is the default when output is to
             a terminal.

     -c      Use time when file status was last changed for sorting (-t) or
             long printing (-l).

     -d      Directories are listed as plain files (not searched recursively).
:
```

MacOS Console



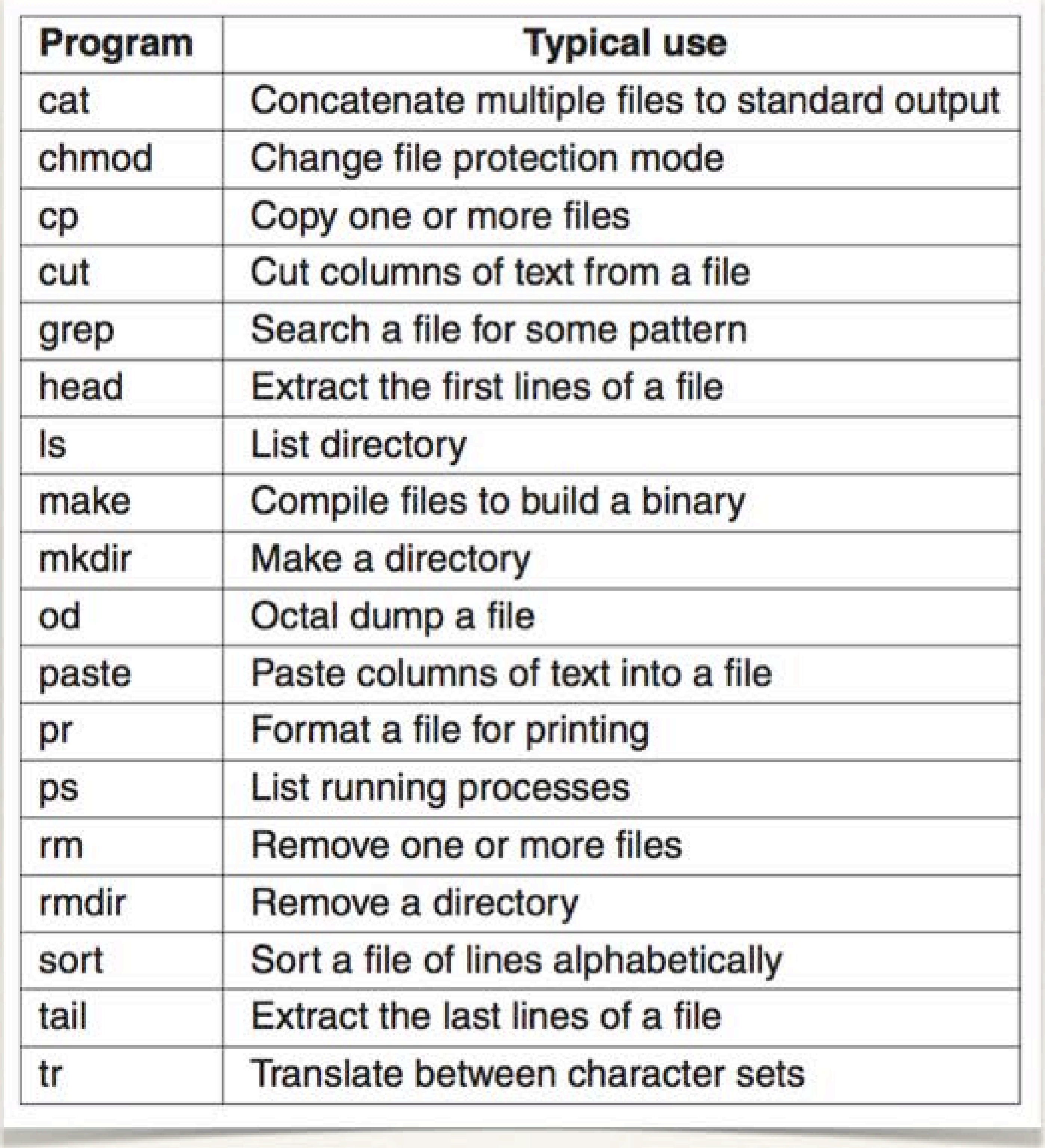
```
                 Megabyte, Gigabyte, Terabyte and Petabyte in order to reduce the
                 number of digits to three or less using base 2 for sizes.

         -i      For each file, print the file's file serial number (inode num-
                 ber).

         -k      If the -s option is specified, print the file size allocation in
                 kilobytes, not blocks.  This option overrides the environment
                 variable BLOCKSIZE.

         -L      Follow all symbolic links to final target and list the file or
                 directory the link references rather than the link itself.  This
                 option cancels the -P option.

         -l      (The lowercase letter ``ell''.)  List in long format.  (See
                 below.)  If the output is to a terminal, a total sum for all the
                 file sizes is output on a line before the long listing.

         -m      Stream output format; list files across the page, separated by
                 commas.

         -n      Display user and group IDs numerically, rather than converting to
                 a user or group name in a long (-l) output.  This option turns on
:
```

MacOS Console



```
Last login: Mon Apr  9 22:06:11 on ttys000
[Fabio-MacBook:~ fabio$ man ls
[Fabio-MacBook:~ fabio$ ls -la
total 28488
drwxr-xr-x@  114 fabio  staff      3876  8 Apr 13:25 .
drwxr-xr-x     6 root   admin       204 21 Dec  2015 ..
-rw-r--r--     1 fabio  staff        16 14 Jun  2016 .7486160831680234
drwxr-xr-x     6 fabio  staff       204 15 Feb  2015 .AllDRMRemoval
-r--------     1 fabio  staff         7  1 Nov 19:19 .CFUserTextEncoding
-rw-r--r--@    1 fabio  staff     65540  9 Apr 22:33 .DS_Store
drwxr-xr-x     3 fabio  staff       102 15 Feb  2015 .Epubor
-rw-r--r--     1 fabio  staff     45993  1 Jan  2016 .Soulseek.1451672273056
-rw-r--r--     1 fabio  staff     45993  2 Jan  2016 .Soulseek.1451675873387
-rw-r--r--     1 fabio  staff     45993  2 Jan  2016 .Soulseek.1451678942645
drwxr-xr-x     3 fabio  staff       102 18 Apr  2017 .SoulseekQt
drwxrwxrwt@    3 fabio  staff       102  8 Apr  2014 .TemporaryItems
drwx------     6 fabio  staff       204  9 Apr 19:33 .Trash
drwxr-xr-x     3 fabio  staff       102 15 Feb  2015 .Ultimate
-rw-------     1 fabio  staff       327 27 Nov 00:45 .Xauthority
drwxr-xr-x     4 fabio  staff       136 14 Nov  2013 .adobe
drwxr-xr-x     3 fabio  staff       102 25 Feb  2017 .anaconda
drwxr-x---     4 fabio  staff       136 12 Jul  2017 .android
drwxr-xr-x     3 fabio  staff       102  5 Mar  2015 .astropy
drwxr-xr-x    15 fabio  staff       510 22 Nov  2016 .atom
```

Other bash commands



Program	Typical use
cat	Concatenate multiple files to standard output
chmod	Change file protection mode
cp	Copy one or more files
cut	Cut columns of text from a file
grep	Search a file for some pattern
head	Extract the first lines of a file
ls	List directory
make	Compile files to build a binary
mkdir	Make a directory
od	Octal dump a file
paste	Paste columns of text into a file
pr	Format a file for printing
ps	List running processes
rm	Remove one or more files
rmdir	Remove a directory
sort	Sort a file of lines alphabetically
tail	Extract the last lines of a file
tr	Translate between character sets

Tanenbaum p730

the pipe line

- It frequently occurs that the first program in a command line produces output that is used as input to the next program. In the above example, we used the file temp to hold this output. However, Linux provides a simpler construction to do the same thing. In

```
sort <in | head –30
```

- the vertical bar, called the pipe symbol, says to take the output from sort and use it as the input to head, eliminating the need for creating, using, and removing the temporary file. A collection of commands connected by pipe symbols, called a pipeline, may contain arbitrarily many commands. A four-component pipeline is shown by the following example:

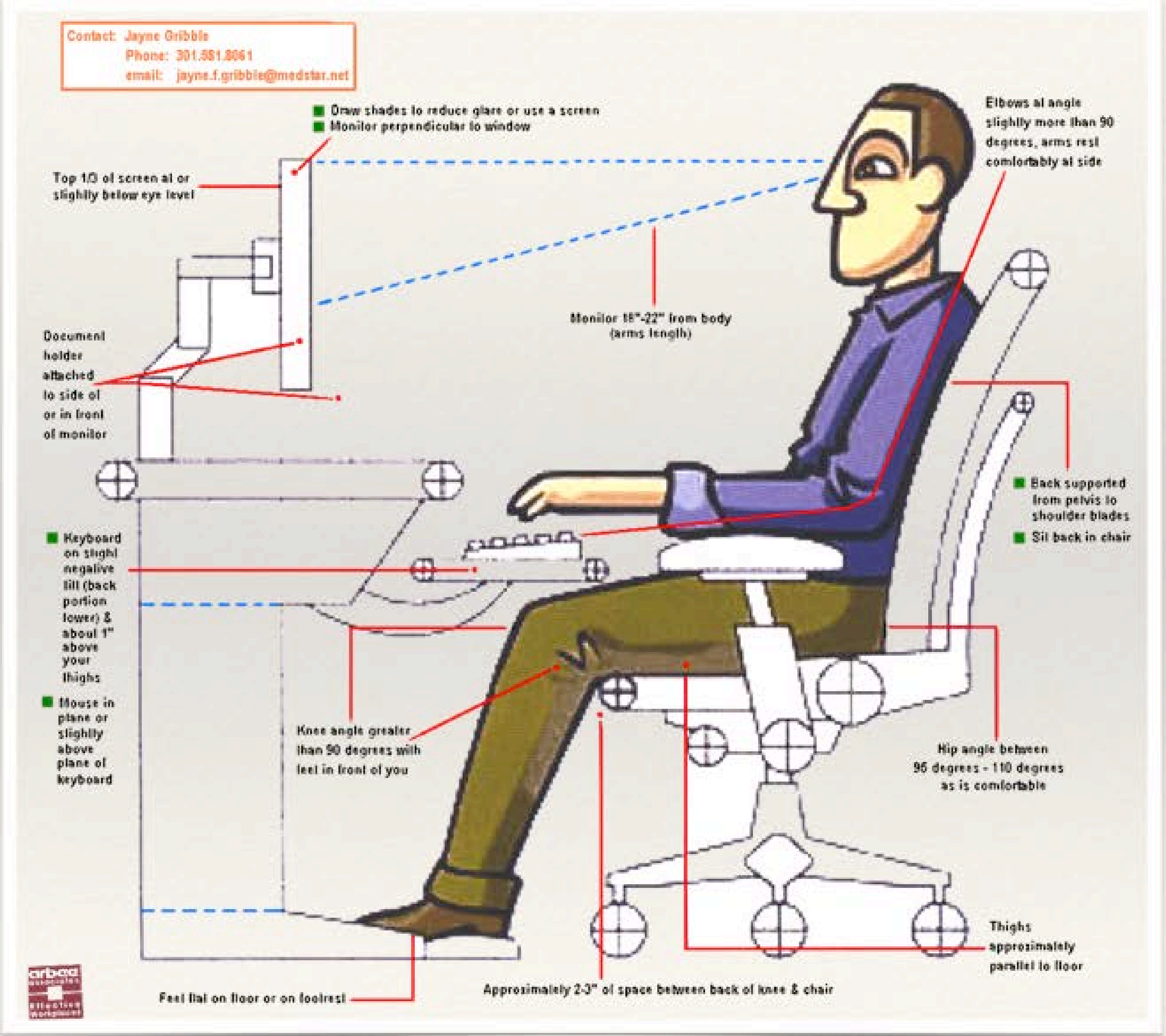
```
grep ter *.t | sort | head –20 | tail –5 >foo
```

- Here all the lines containing the string ''ter'' in all the files ending in .t are written to standard output, where they are sorted. The first 20 of these are selected out by head, which passes them to tail, which writes the last five (i.e., lines 16 to 20 in the sorted list) to foo. This is an example of how Linux provides basic building blocks (numerous filters), each of which does one job, along with a mechanism for them to be put together in almost limitless ways.

Tanenbaum p728

GUIs

Ergonomics



Contact: Jayne Gribble
Phone: 301.581.8061
email: jayne.f.gribble@medstar.net

- 🟩 Draw shades to reduce glare or use a screen
- 🟩 Monitor perpendicular to window

Top 1/3 of screen at or slightly below eye level

Monitor 18"-22" from body (arms length)

Elbows at angle slightly more than 90 degrees, arms rest comfortably at side

Document holder attached to side of or in front of monitor

- 🟩 Back supported from pelvis to shoulder blades
- 🟩 Sit back in chair

- 🟩 Keyboard on slight negative tilt (back portion lower) & about 1" above your thighs
- 🟩 Mouse in plane or slightly above plane of keyboard

Knee angle greater than 90 degrees with feet in front of you

Hip angle between 95 degrees - 110 degrees as is comfortable

Thighs approximately parallel to floor

Feet flat on floor or on footrest

Approximately 2-3" of space between back of knee & chair

Human-Computer Interaction

- Human-computer interaction (HCI) can be defined in many possible ways.
    - «Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.»
    - «"Human-computer interaction " is, put simply, the study of people, computer technology and the ways these influence each other. We study HCI to determine how we can make this computer technology more usable by people.»
    - «Human-computer interaction is about designing computer sys at support op so that they can car out their activities productively and safely.»
- The computer is a tool, a complex artifact that can extend our reach. The design discipline of human-computer interaction systematically applies knowledge about human purposes, human capabilities and limitations, and machine capabilities and limitations in order to enable us to do things that we could not do before. Another goal of HCI, as suggested in the definitions given above, is to **enhance the quality of the interaction between people and computers**. We strive, for example, to make technology **easier for people to learn** and **easier for them to use**.

Baecker p1

Human-Computer Interaction

- Donald A. Norman, in his book **The Psychology of Everyday Things** (1988), examines the design of a multitude of common objects, such as doors, light switches, water faucets, stove controls, refrigerators, washer–dryers, slide projectors, telephones, and watches. He finds examples of bad design almost everywhere, even for objects far simpler than most human-computer interfaces. Norman then introduces several concepts that he uses in his analyses of both good and bad design:
  - **Affordances** are the perceived properties of an artifact that determine how it could possibly be used. For example, buttons are for pushing, menus are for choosing.
  - **Constraints** are physical, semantic, cultural, and logical factors that encourage proper actions and prevent erroneous ones.
  - **Conceptual models** are mental models of a system which allow users to understand the system, to predict the effects of their actions, and to interpret the results.
  - **Mappings** describe the relationships between controls and their effects on a system. For example, moving a control to the left should move a corresponding display object left.
  - **Visibility** in the design of a system makes apparent to users the conceptual model of the system and the actions they are allowed to take.
  - **Feedback** from a system provides information about the effects of users' actions.

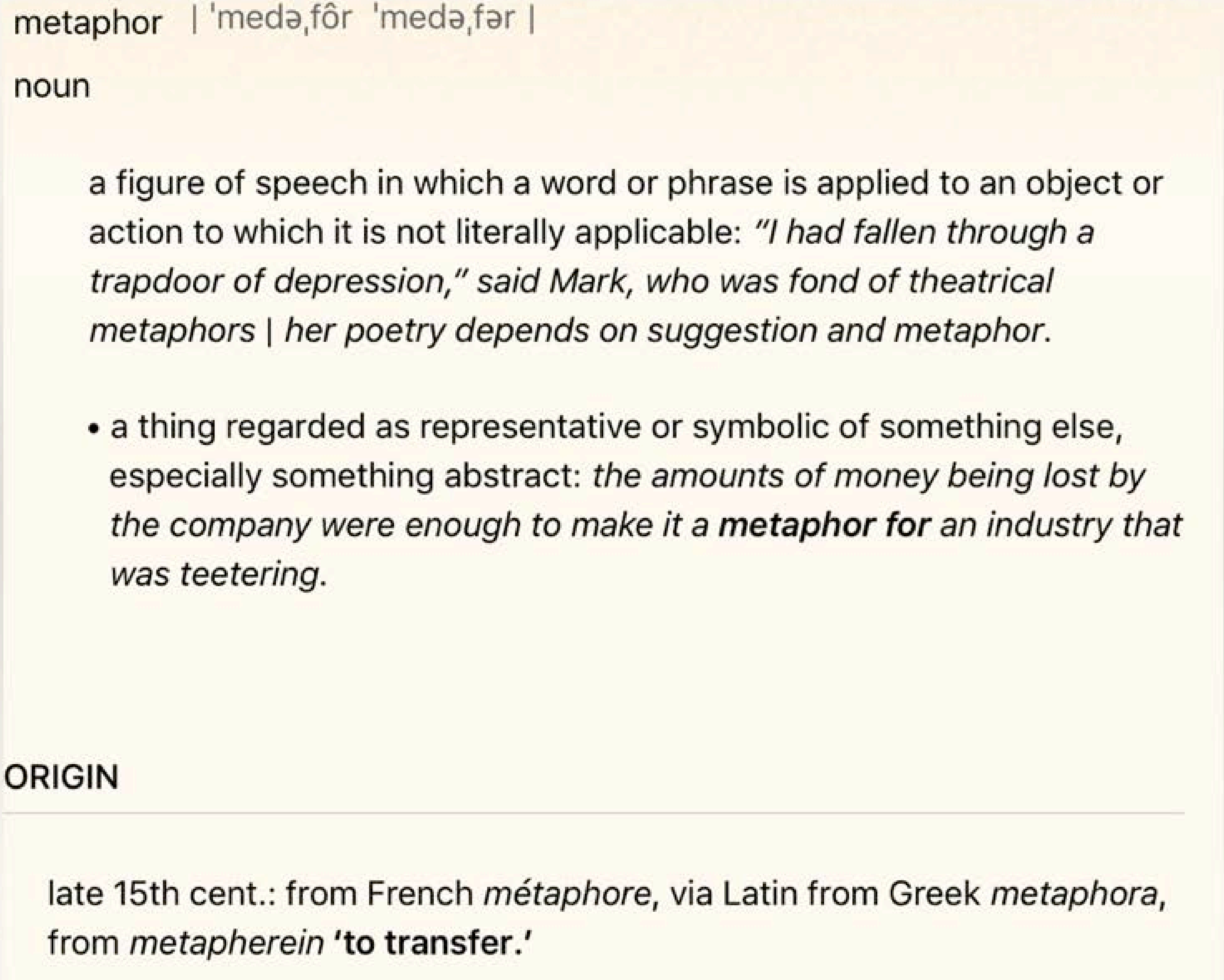
Baecker p1

Human-Computer Interaction

- In the book **The Psychology of Everyday Things** (1988), Donald A. Norman uses the six key concepts defined above in his **prescriptions for user-centered design**:
  - Make it easy to determine **what actions are possible** at any moment (make use of constraints).
  - Make things **visible**, including the conceptual model of the system, the alternative actions, and the results of actions.
  - Make it easy to evaluate **the current state of the system.**
  - Follow **natural mappings** between **intentions** and e **required actions**; between **actions** and e **resulting effect**; and between the **information** that is visible and the interpretation of **the system state**.
    In other words, make sure that (1) the user can figure out **what to do**, and (2) the user can tell **what is going on**.

Baecker p1

Metaphors



metaphor | ˈmedəˌfôr ˈmedəˌfər |

noun

a figure of speech in which a word or phrase is applied to an object or action to which it is not literally applicable: *"I had fallen through a trapdoor of depression," said Mark, who was fond of theatrical metaphors | her poetry depends on suggestion and metaphor.*

- a thing regarded as representative or symbolic of something else, especially something abstract: *the amounts of money being lost by the company were enough to make it a* ***metaphor for*** *an industry that was teetering.*

ORIGIN

late 15th cent.: from French *métaphore*, via Latin from Greek *metaphora*, from *metapherein* **'to transfer.'**

The desktop metaphor

- In computing, the desktop metaphor is an interface metaphor which is a set of unifying concepts used by graphical user interfaces to help users interact more easily with the computer. The desktop metaphor treats the computer monitor as if it is the user's desktop, upon which objects such as documents and folders of documents can be placed. A document can be opened into a window, which represents a paper copy of the document placed on the desktop. Small applications called desk accessories are also available, such as a desk calculator or notepad, etc.

- The desktop metaphor itself has been extended and stretched with various implementations of desktop environments, since access to features and usability of the computer are usually more important than maintaining the 'purity' of the metaphor. Hence we find trash cans on the desktop, as well as disks and network volumes (which can be thought of as filing cabinets—not something normally found on a desktop). Other features such as menu bars, task bars, or docks have no counterpart on a real-world desktop.

Wikipedia

Computer Metaphors

- When working on Linux systems through a graphical interface, users may use mouse clicks to run applications or open files, drag and drop to copy files from one location to another, and so on. In addition, users may invoke a terminal emulator program, or xterm, which provides them with the basic command-line interface to the operating system.

📖 Tanenbaum p726

Computer Metaphors

- The GUI for Linux is similar to the first GUIs developed for UNIX systems in the 1970s, and popularized by Macintosh and later Windows for PC platforms. The GUI creates a **desktop environment**, a familiar **metaphor** with windows, icons, folders, toolbars, and drag-and-drop capabilities. A full desktop environment contains a window manager, which controls the placement and appearance of windows, as well as various applications, and provides a consistent **graphical interface**. Popular desktop environments for Linux include GNOME (GNU Network Object Model Environment) and KDE (K Desktop Environment).

📖 Tanenbaum p725

Computer Metaphors

DELETE COOKIES?!
gegen-den-strich.com
FERNANDEZ

Open Source Software

Free Software

- The **free software movement** (FSM) or free/open source software movement (FOSSM) or free/libre open source software (FLOSS) is a social movement with the goal of obtaining and guaranteeing certain freedoms for software users, namely the freedom to run the software, to study and change the software, and to redistribute copies with or without changes.
- Regarding the meaning and misunderstandings of the word free, those who work within the free software camp have searched for less ambiguous terms and analogies like "**free beer vs free speech**" in efforts to convey the intended semantics, so that there is no confusion concerning the profitability of free software.
- The two most prominent people associated with the movement, Richard Stallman and Linus Torvalds,

Wikipedia

Richard Stallman

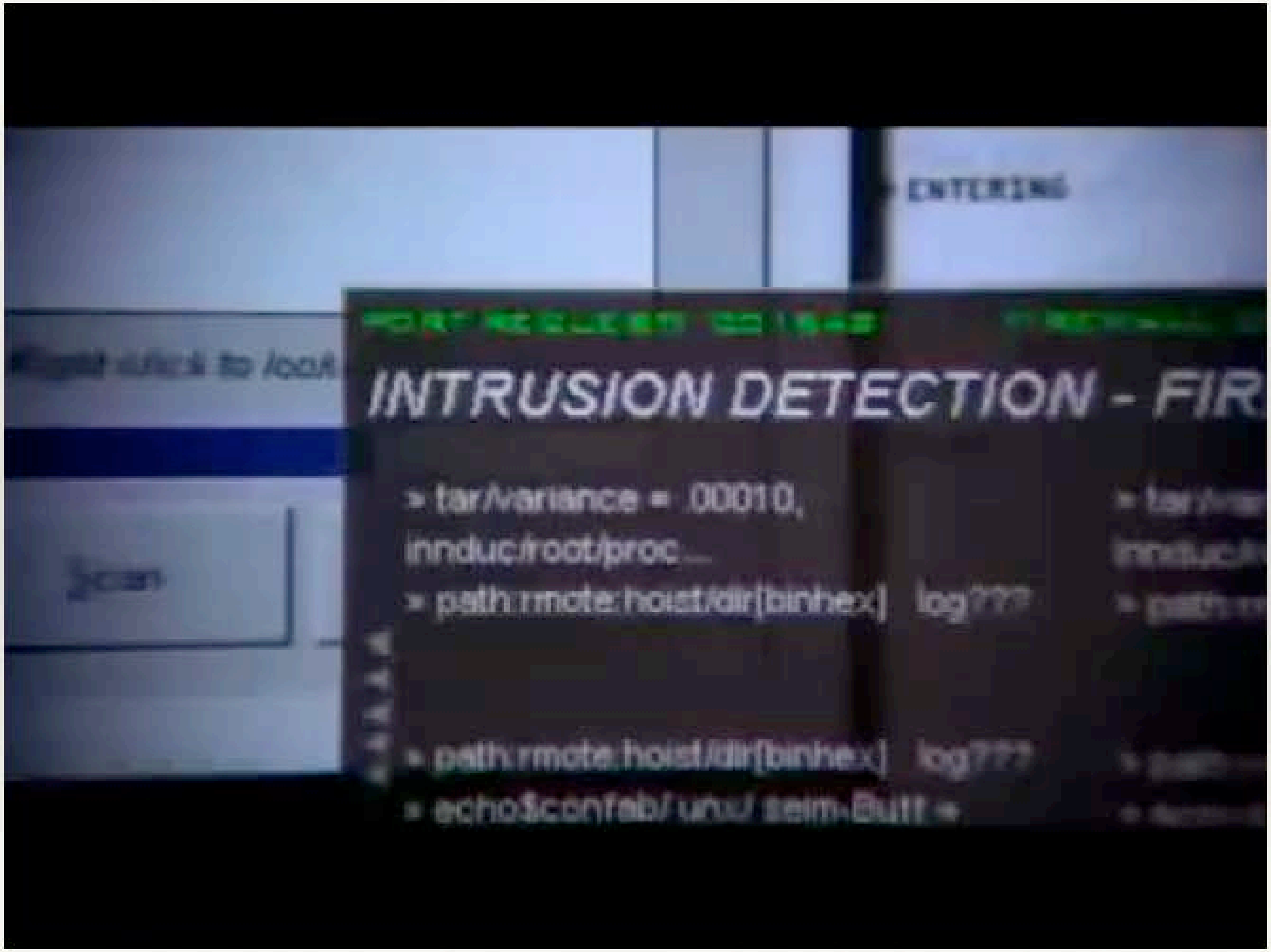
GNU project

Linus Torvalds

Linux project

Computer interfacing in popular culture

two on keyboard



```
> tar/variance = .00010,
innduc/root/proc...
> path:rmote:hoist/dir[binhex]   log???

> path:rmote:hoist/dir[binhex]   log???
> echo$confab/ unx/ seim-Buff =
```

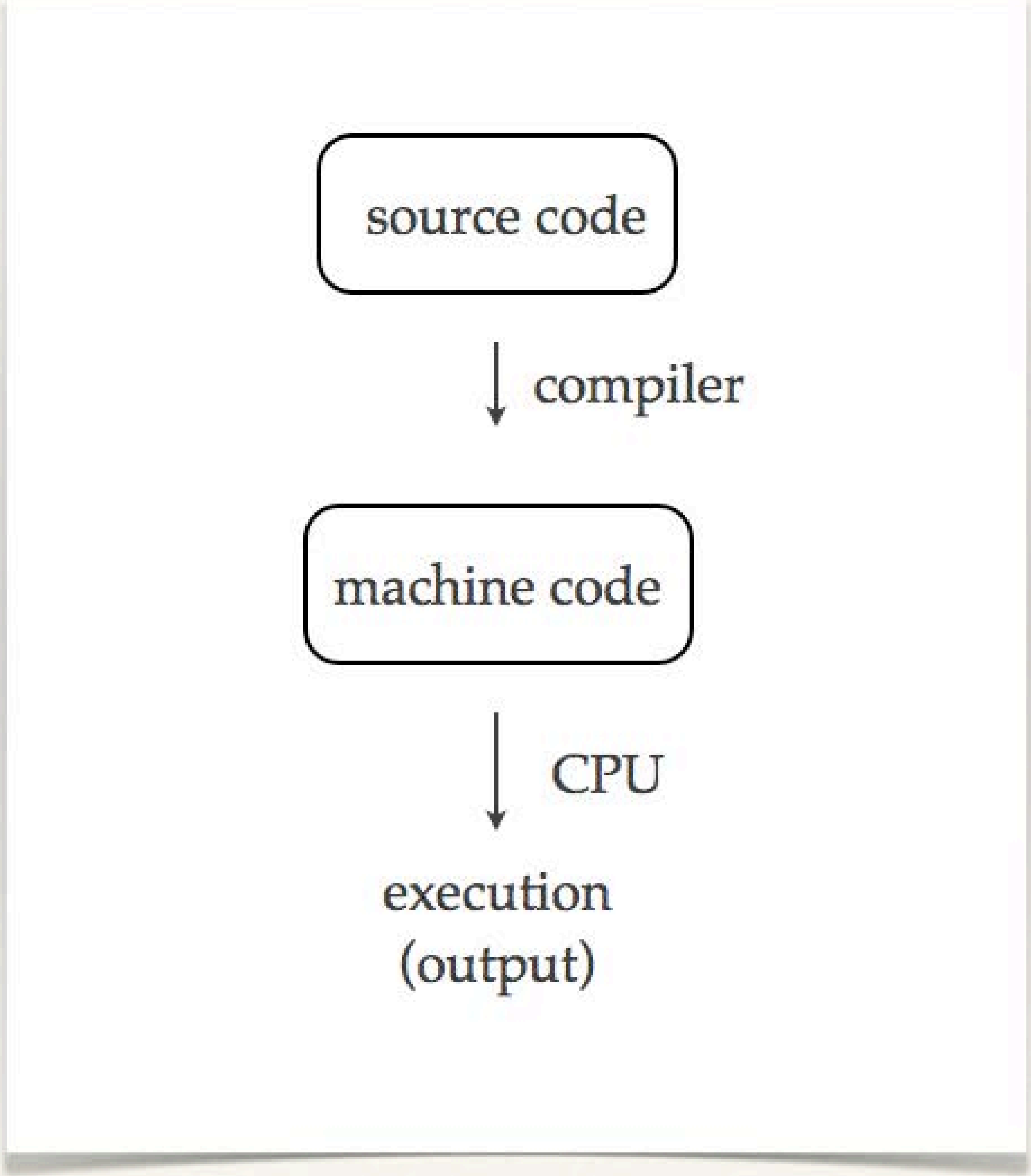
user interface

user interface

compiled vs interpreted

compiled

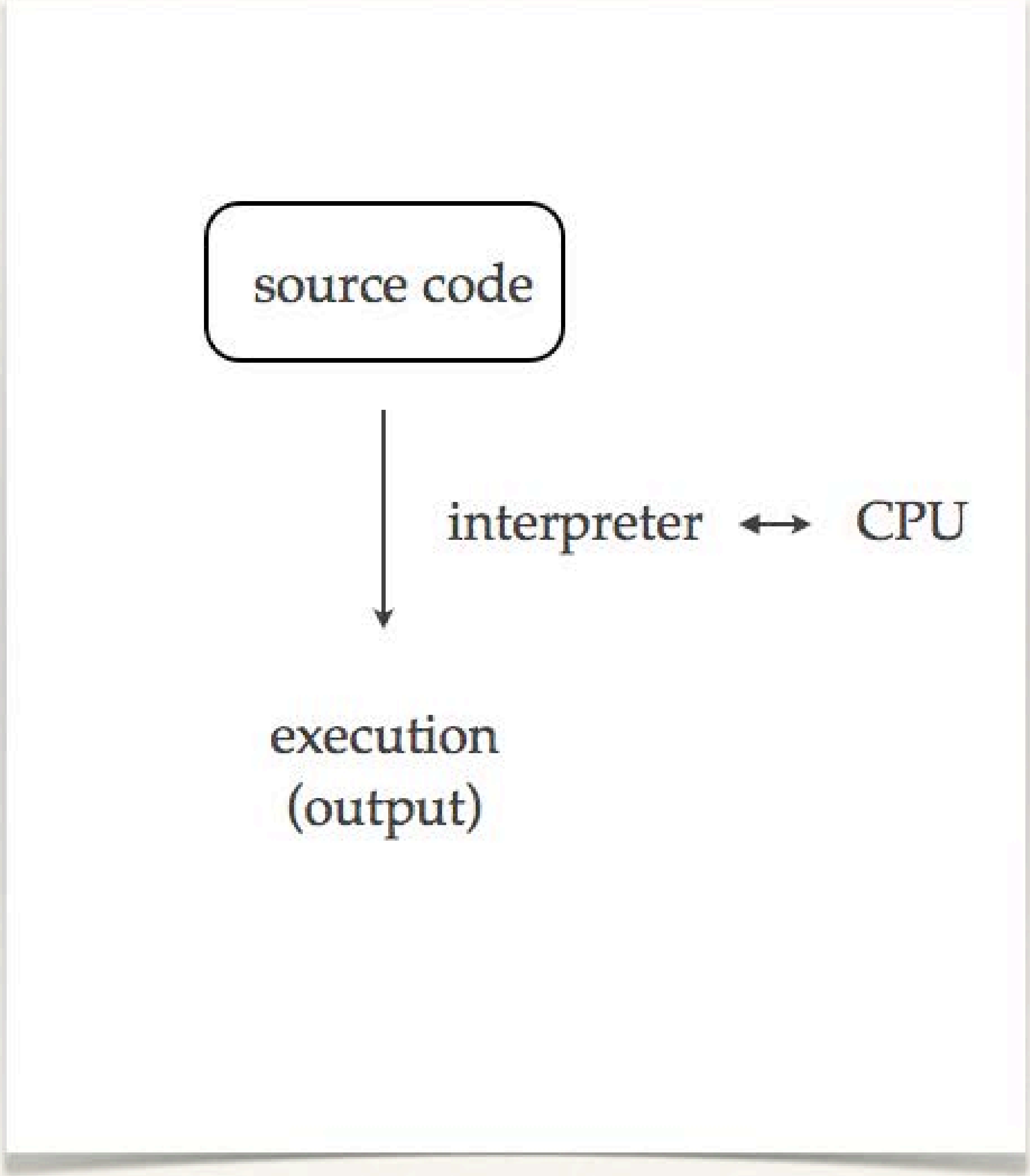
- PROs
  - faster
  - it is compiled only once
- CONs
  - needs an intermediate step (machine code)
  - executes all the instructions at once



Fox p25

interpreted

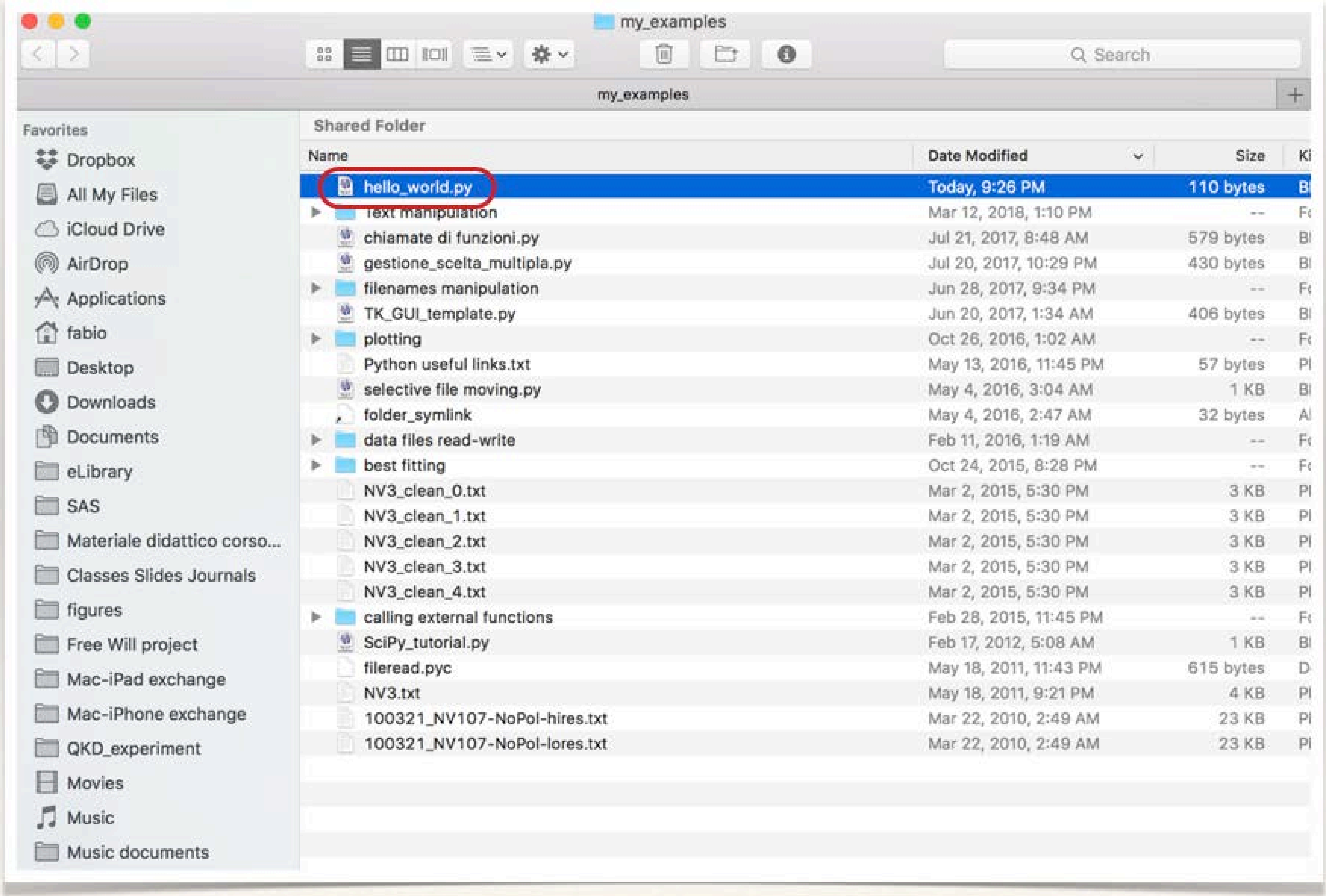
- PROs
  - no intermediate step
  - execution step-by-step
- CONs
  - it is slower
  - needs to “compile” each time



Fox p95

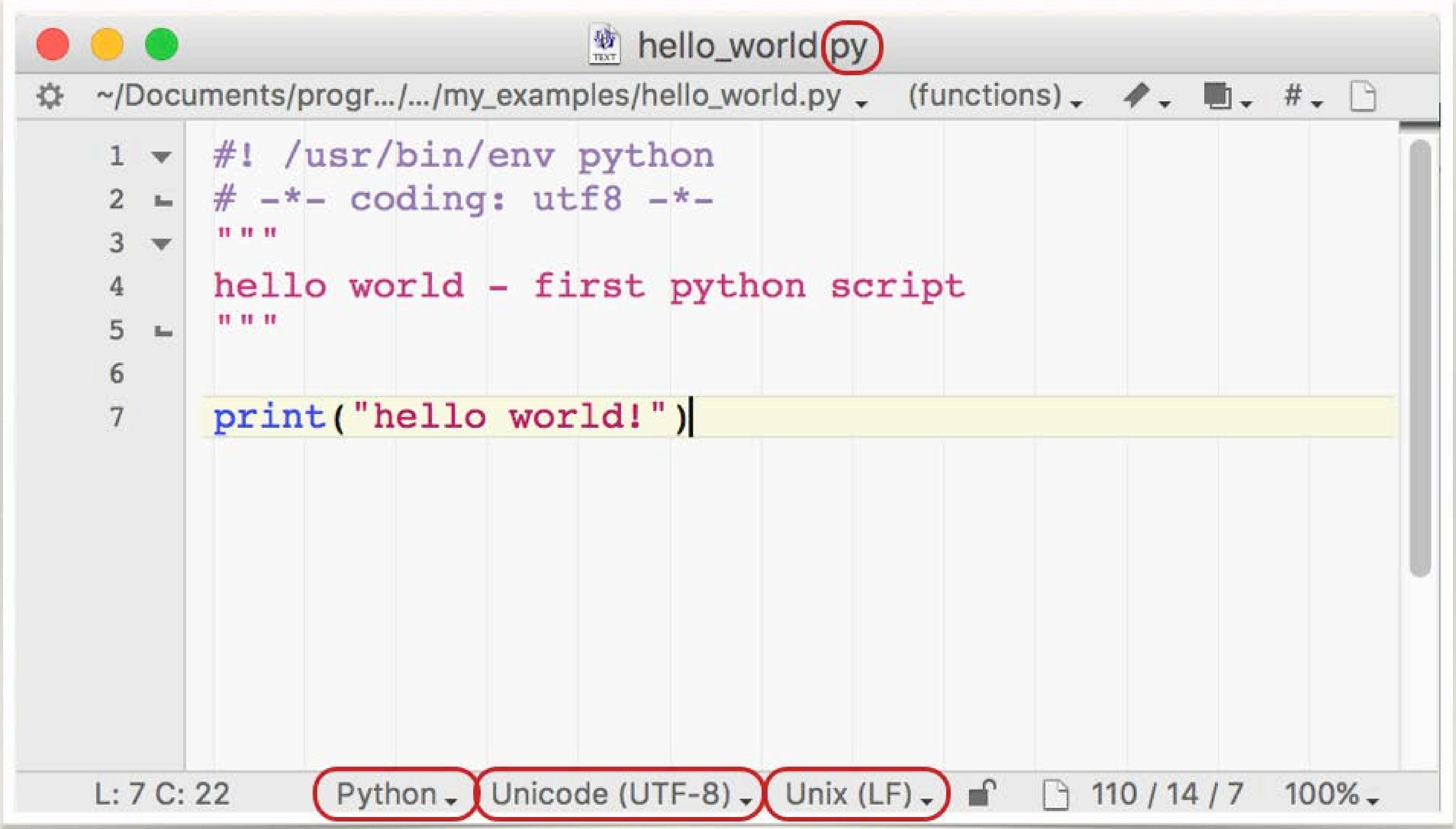
interpreted example (python)

our first interpreted program



Name	Date Modified	Size	Ki
hello_world.py	Today, 9:26 PM	110 bytes	B
Text manipulation	Mar 12, 2018, 1:10 PM	--	Fo
chiamate di funzioni.py	Jul 21, 2017, 8:48 AM	579 bytes	Bl
gestione_scelta_multipla.py	Jul 20, 2017, 10:29 PM	430 bytes	Bl
filenames manipulation	Jun 28, 2017, 9:34 PM	--	Fo
TK_GUI_template.py	Jun 20, 2017, 1:34 AM	406 bytes	Bl
plotting	Oct 26, 2016, 1:02 AM	--	Fo
Python useful links.txt	May 13, 2016, 11:45 PM	57 bytes	Pl
selective file moving.py	May 4, 2016, 3:04 AM	1 KB	Bl
folder_symlink	May 4, 2016, 2:47 AM	32 bytes	Al
data files read-write	Feb 11, 2016, 1:19 AM	--	Fo
best fitting	Oct 24, 2015, 8:28 PM	--	Fo
NV3_clean_0.txt	Mar 2, 2015, 5:30 PM	3 KB	Pl
NV3_clean_1.txt	Mar 2, 2015, 5:30 PM	3 KB	Pl
NV3_clean_2.txt	Mar 2, 2015, 5:30 PM	3 KB	Pl
NV3_clean_3.txt	Mar 2, 2015, 5:30 PM	3 KB	Pl
NV3_clean_4.txt	Mar 2, 2015, 5:30 PM	3 KB	Pl
calling external functions	Feb 28, 2015, 11:45 PM	--	Fo
SciPy_tutorial.py	Feb 17, 2012, 5:08 AM	1 KB	Bl
fileread.pyc	May 18, 2011, 11:43 PM	615 bytes	D
NV3.txt	May 18, 2011, 9:21 PM	4 KB	Pl
100321_NV107-NoPol-hires.txt	Mar 22, 2010, 2:49 AM	23 KB	Pl
100321_NV107-NoPol-lores.txt	Mar 22, 2010, 2:49 AM	23 KB	Pl

hello world!



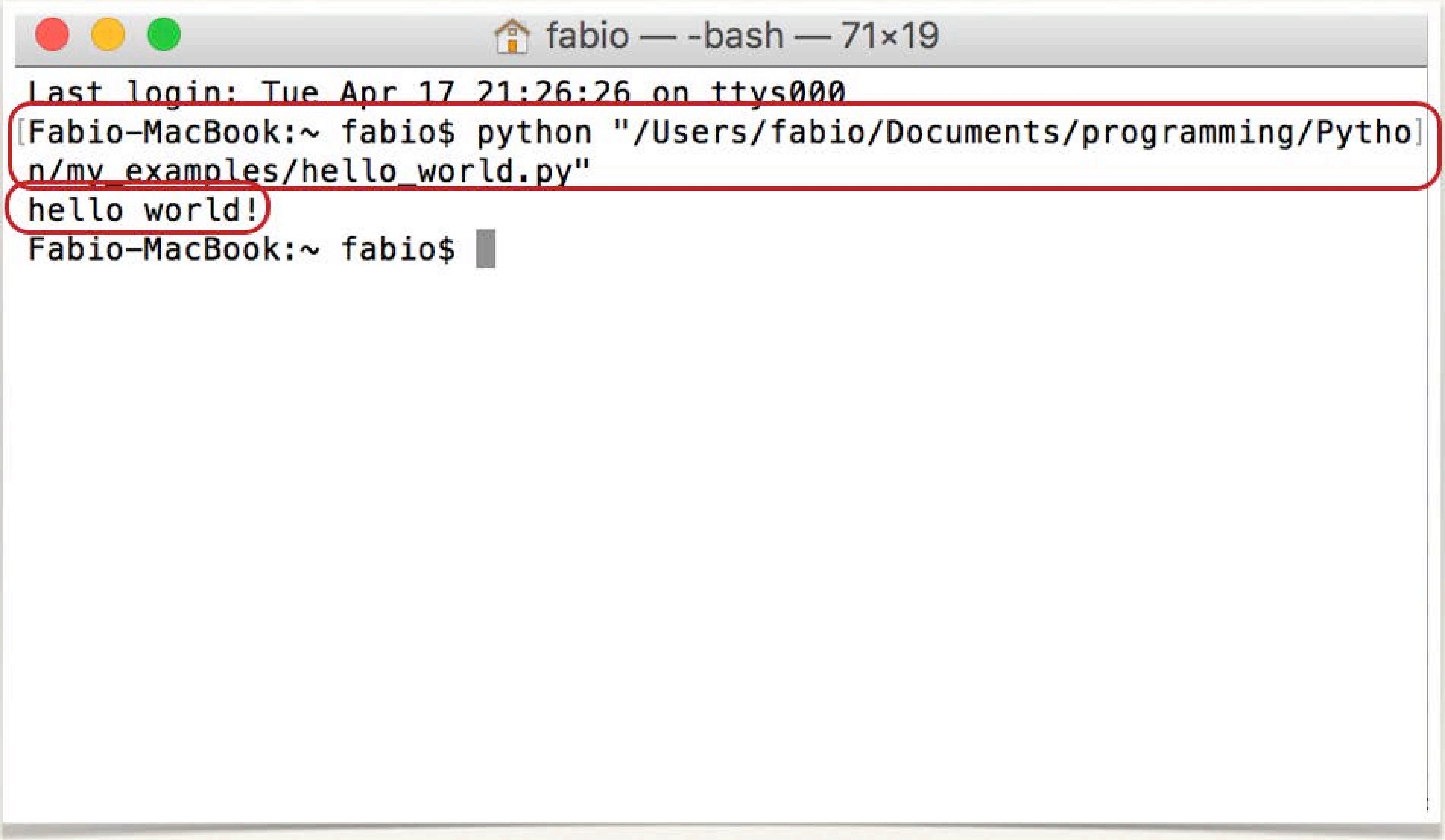
hello_world.py

~/Documents/progr.../.../my_examples/hello_world.py

```
#! /usr/bin/env python
# -*- coding: utf8 -*-
"""
hello world - first python script
"""

print("hello world!")
```

L: 7 C: 22   Python   Unicode (UTF-8)   Unix (LF)   110 / 14 / 7   100%

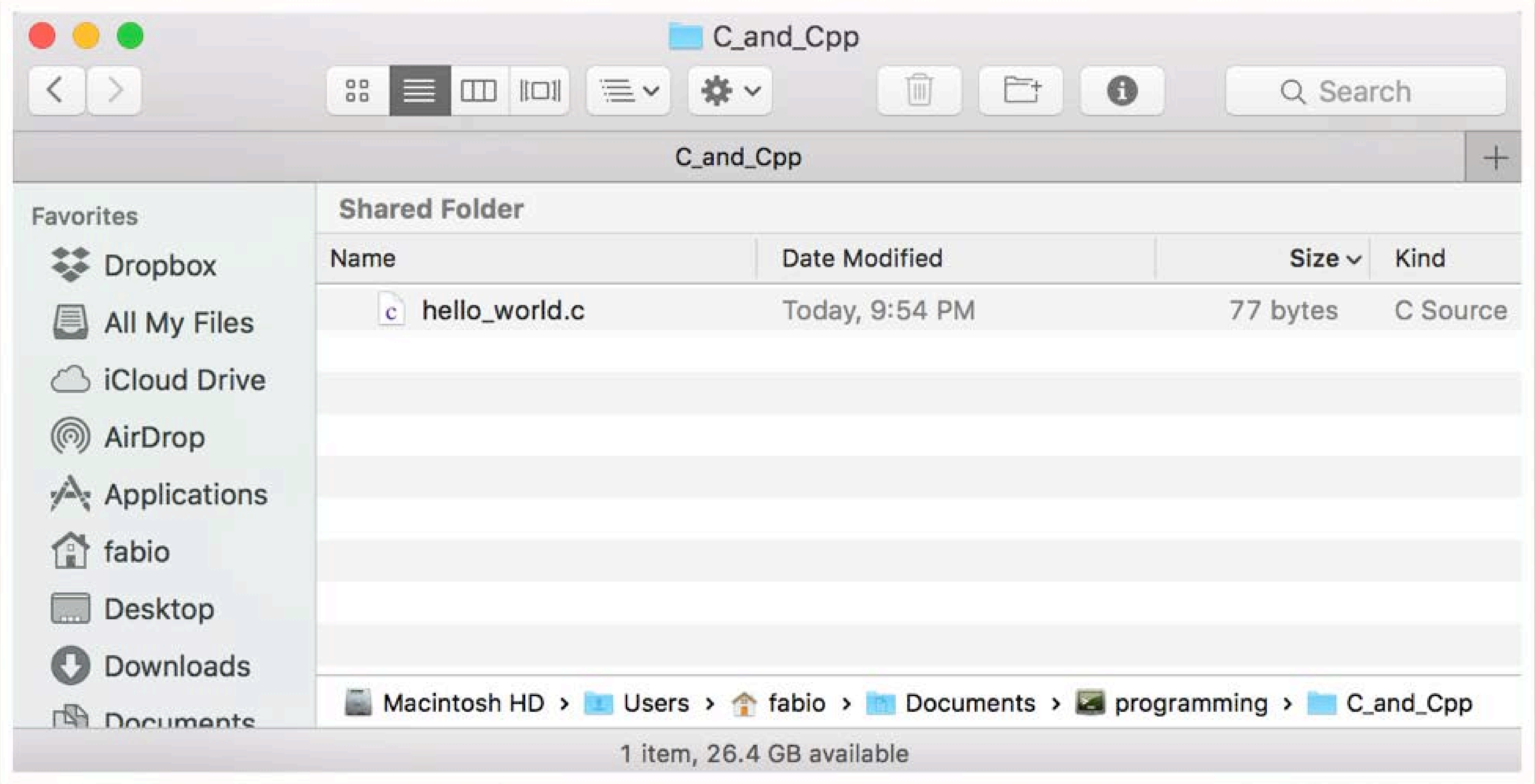
our first output in python!



```
Last login: Tue Apr 17 21:26:26 on ttys000
Fabio-MacBook:~ fabio$ python "/Users/fabio/Documents/programming/Pytho
n/my_examples/hello_world.py"
hello world!
Fabio-MacBook:~ fabio$ 
```

compiled example (C)

source code is a text file

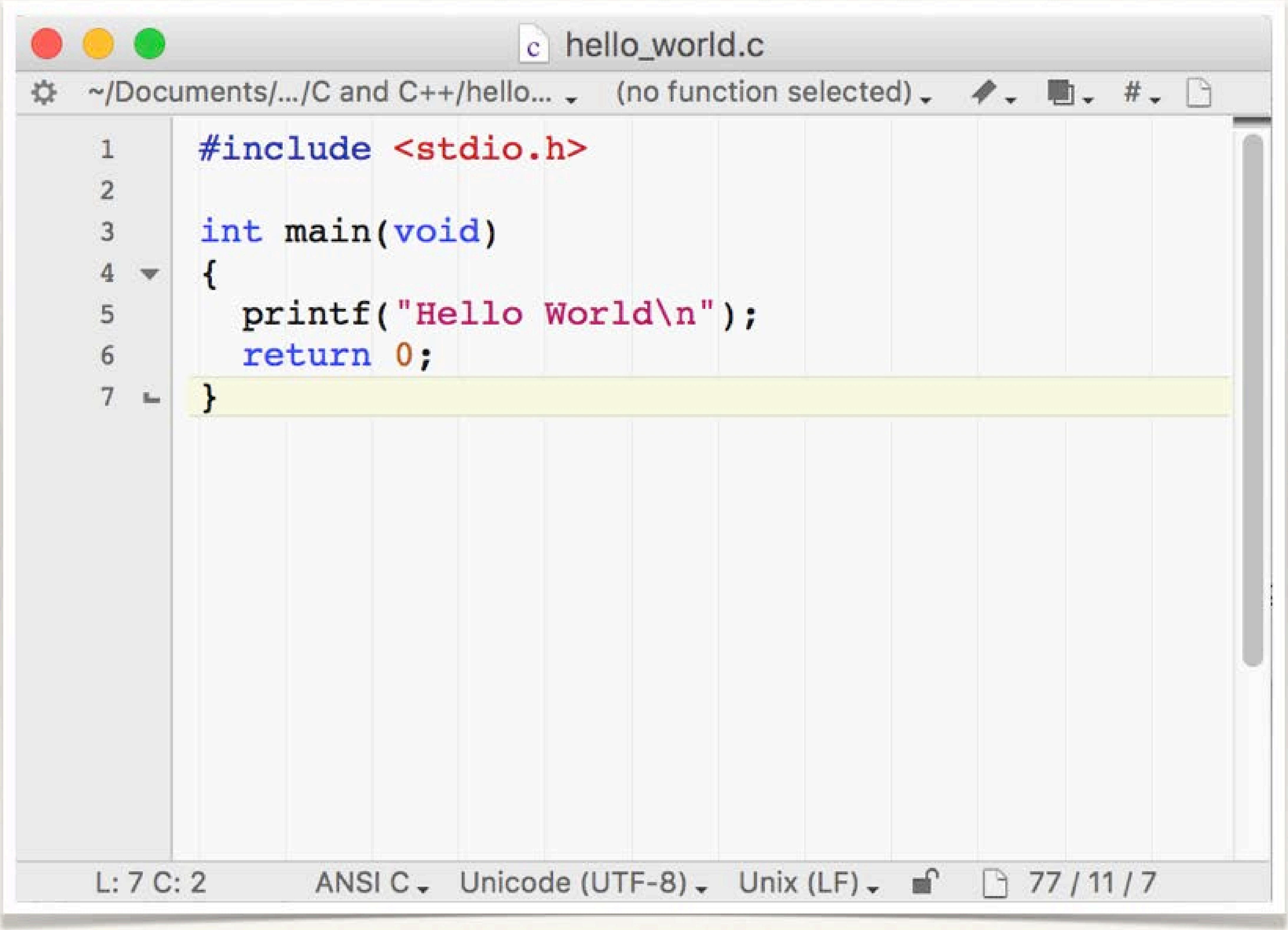


Name	Date Modified	Size	Kind
hello_world.c	Today, 9:54 PM	77 bytes	C Source

Macintosh HD › Users › fabio › Documents › programming › C_and_Cpp

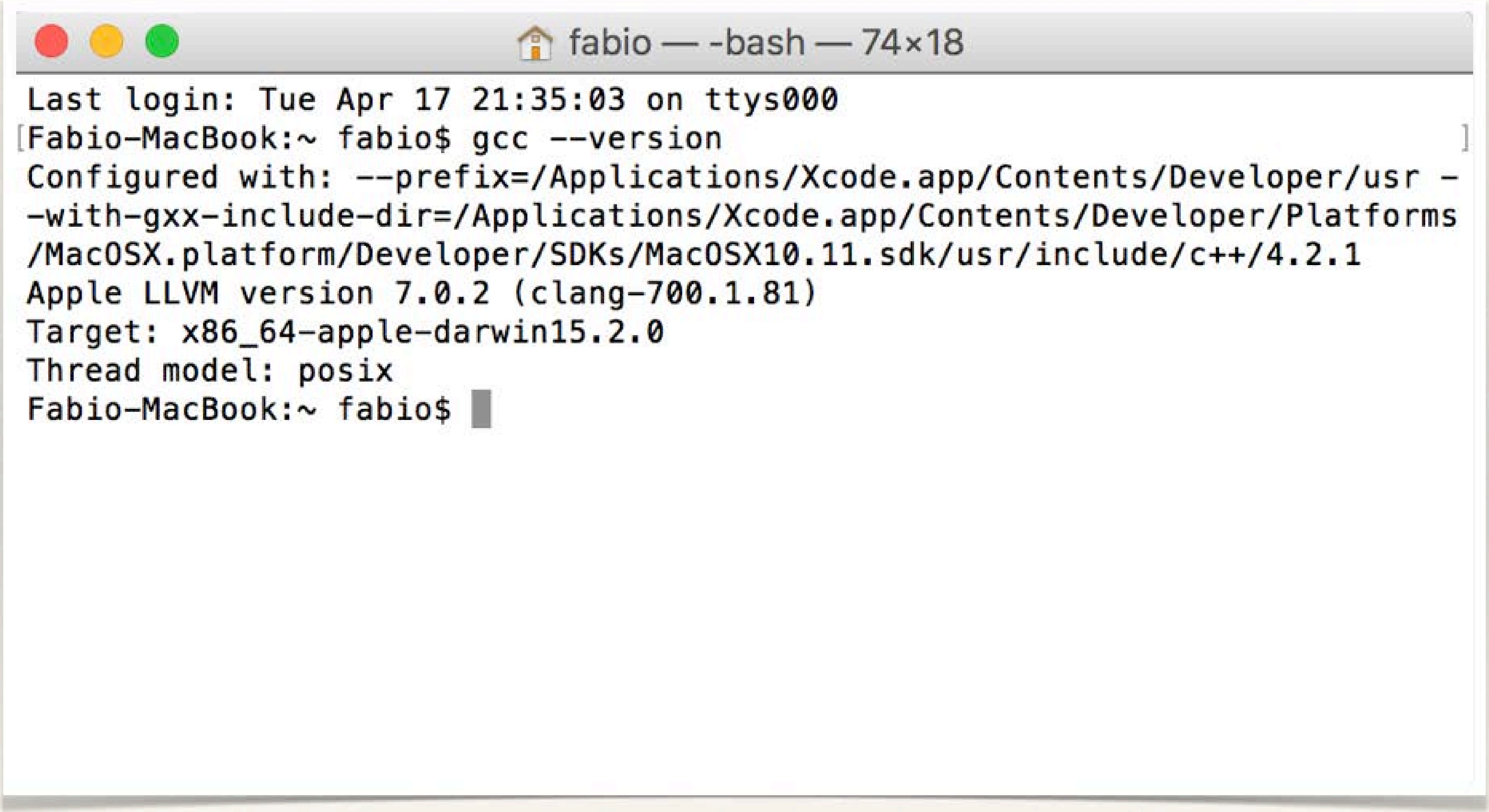
1 item, 26.4 GB available

source code opened in text editor



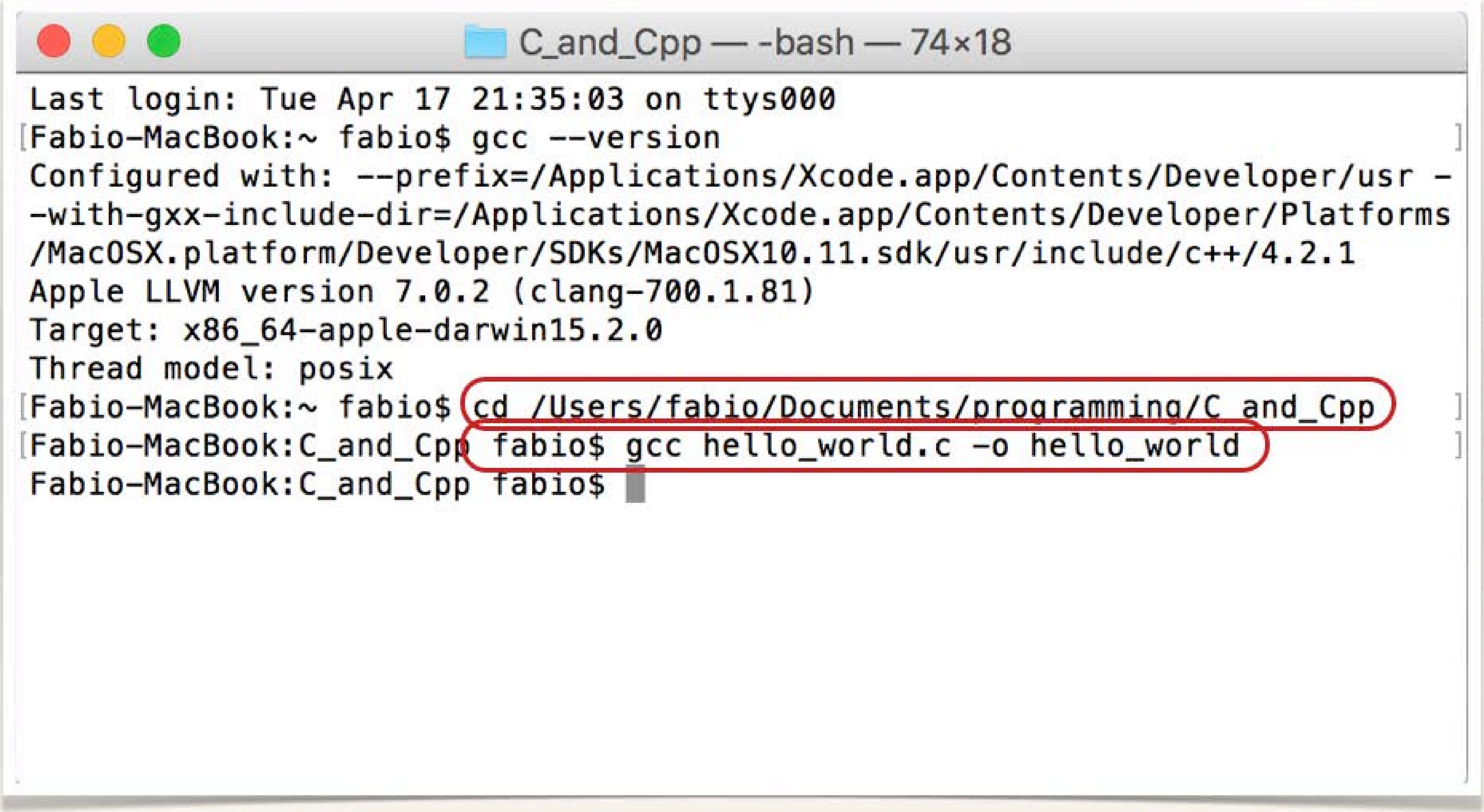
```c
#include <stdio.h>

int main(void)
{
  printf("Hello World\n");
  return 0;
}
```

check our compiler (gcc)



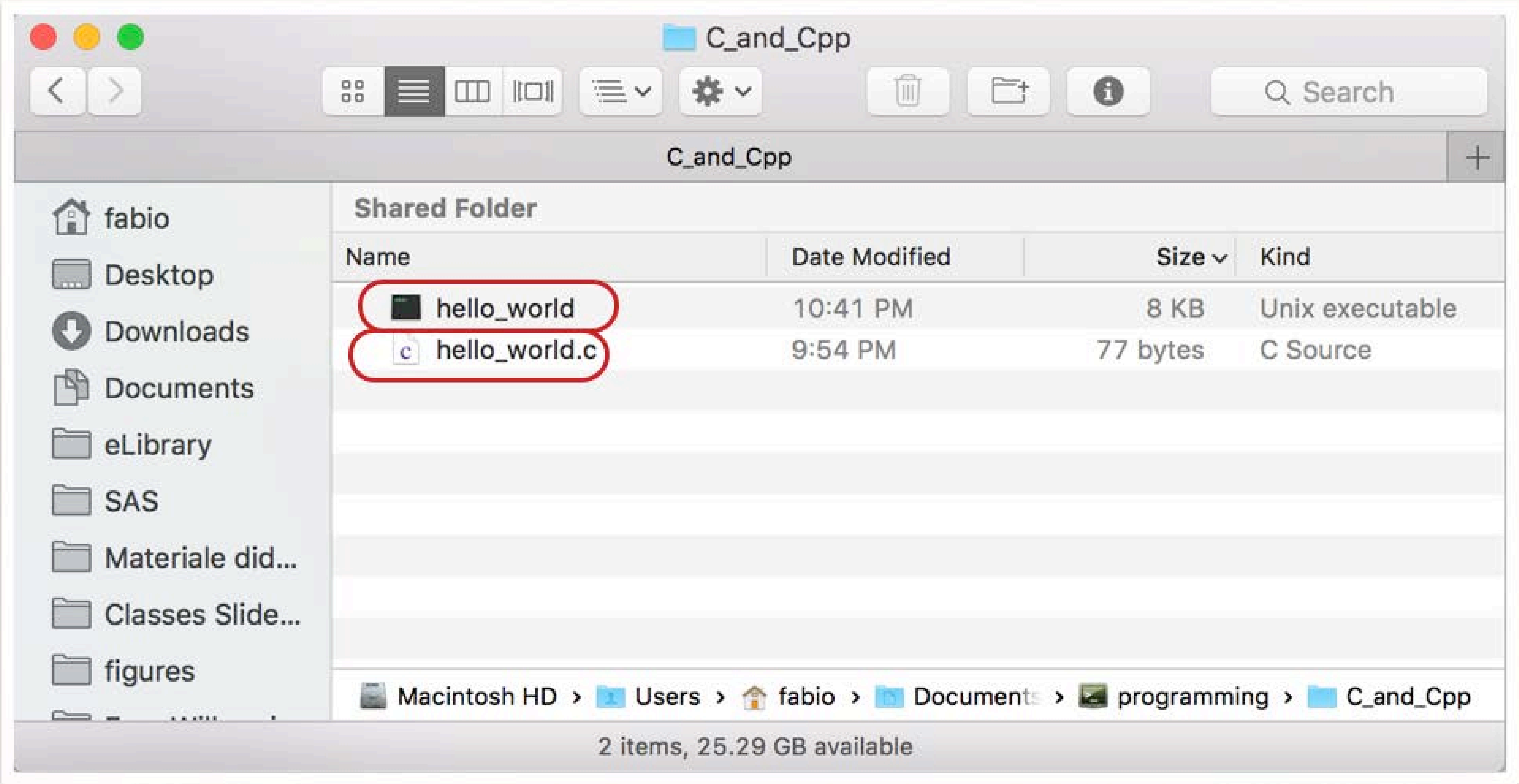
```
Last login: Tue Apr 17 21:35:03 on ttys000
Fabio-MacBook:~ fabio$ gcc --version
Configured with: --prefix=/Applications/Xcode.app/Contents/Developer/usr -
-with-gxx-include-dir=/Applications/Xcode.app/Contents/Developer/Platforms
/MacOSX.platform/Developer/SDKs/MacOSX10.11.sdk/usr/include/c++/4.2.1
Apple LLVM version 7.0.2 (clang-700.1.81)
Target: x86_64-apple-darwin15.2.0
Thread model: posix
Fabio-MacBook:~ fabio$ 
```

CD and compile



```
Last login: Tue Apr 17 21:35:03 on ttys000
Fabio-MacBook:~ fabio$ gcc --version
Configured with: --prefix=/Applications/Xcode.app/Contents/Developer/usr -
-with-gxx-include-dir=/Applications/Xcode.app/Contents/Developer/Platforms
/MacOSX.platform/Developer/SDKs/MacOSX10.11.sdk/usr/include/c++/4.2.1
Apple LLVM version 7.0.2 (clang-700.1.81)
Target: x86_64-apple-darwin15.2.0
Thread model: posix
Fabio-MacBook:~ fabio$ cd /Users/fabio/Documents/programming/C_and_Cpp
Fabio-MacBook:C_and_Cpp fabio$ gcc hello_world.c -o hello_world
Fabio-MacBook:C_and_Cpp fabio$
```

check that we have executable

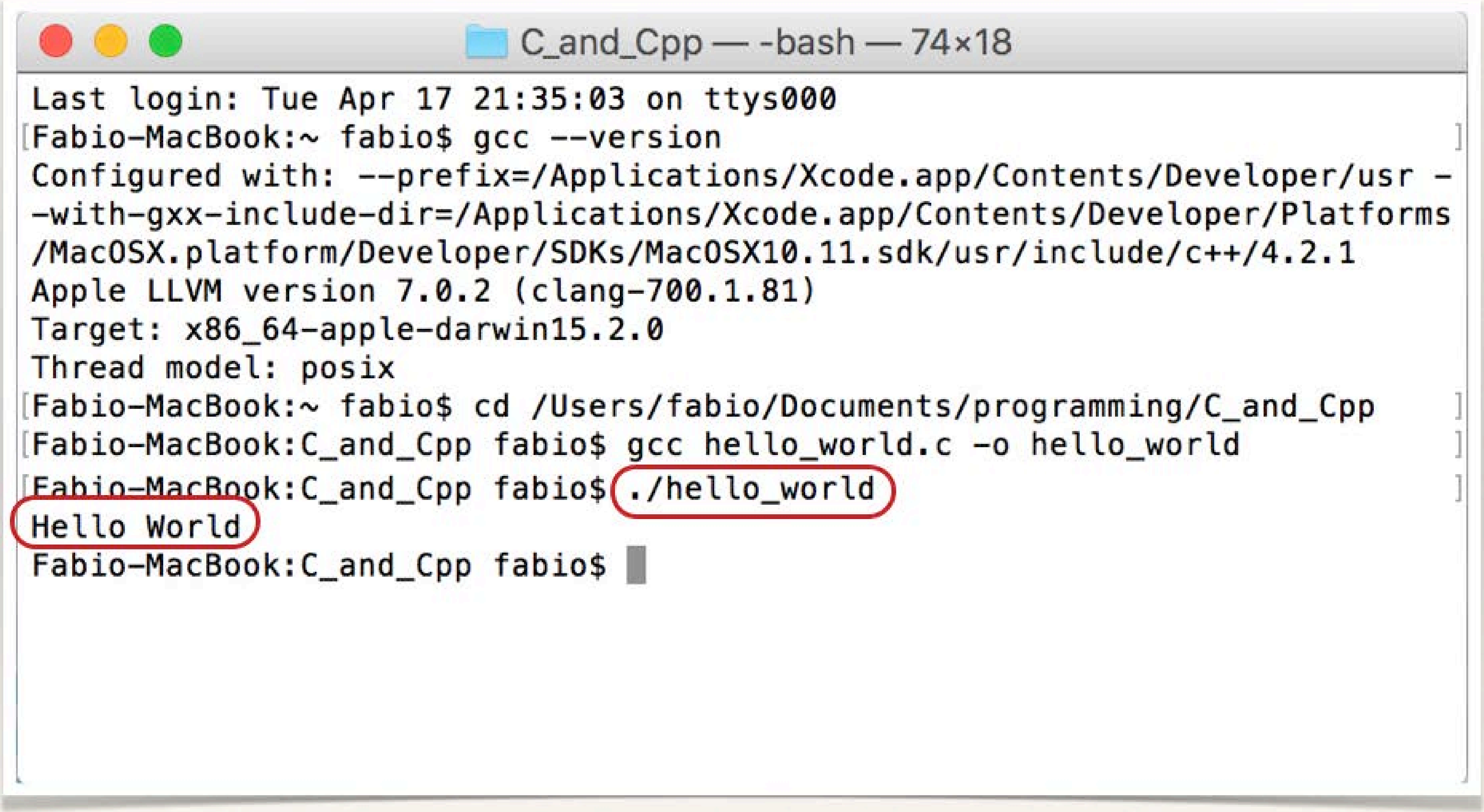


Name	Date Modified	Size	Kind
hello_world	10:41 PM	8 KB	Unix executable
hello_world.c	9:54 PM	77 bytes	C Source

Macintosh HD › Users › fabio › Documents › programming › C_and_Cpp

2 items, 25.29 GB available

execute



```
Last login: Tue Apr 17 21:35:03 on ttys000
Fabio-MacBook:~ fabio$ gcc --version
Configured with: --prefix=/Applications/Xcode.app/Contents/Developer/usr -
-with-gxx-include-dir=/Applications/Xcode.app/Contents/Developer/Platforms
/MacOSX.platform/Developer/SDKs/MacOSX10.11.sdk/usr/include/c++/4.2.1
Apple LLVM version 7.0.2 (clang-700.1.81)
Target: x86_64-apple-darwin15.2.0
Thread model: posix
Fabio-MacBook:~ fabio$ cd /Users/fabio/Documents/programming/C_and_Cpp
Fabio-MacBook:C_and_Cpp fabio$ gcc hello_world.c -o hello_world
Fabio-MacBook:C_and_Cpp fabio$ ./hello_world
Hello World
Fabio-MacBook:C_and_Cpp fabio$ 
```

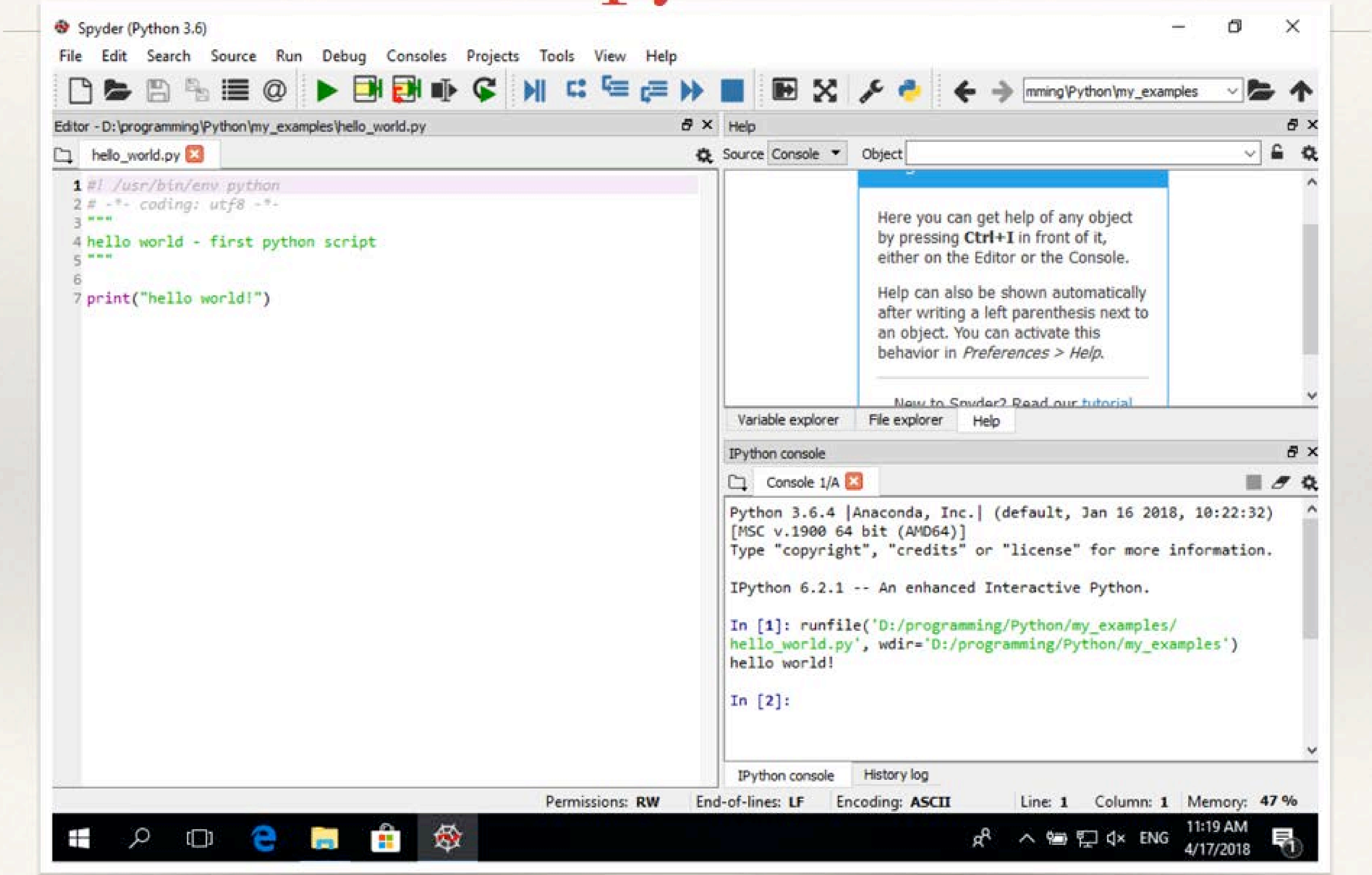
To who we write?

Programming environment

IDE

- IDE = Integrated Development Environment
- “Spyder” is the IDE for python

Spyder



```python
#! /usr/bin/env python
# -*- coding: utf8 -*-
"""
hello world - first python script
"""

print("hello world!")
```

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
Python 3.6.4 |Anaconda, Inc.| (default, Jan 16 2018, 10:22:32)
[MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.2.1 -- An enhanced Interactive Python.

In [1]: runfile('D:/programming/Python/my_examples/
hello_world.py', wdir='D:/programming/Python/my_examples')
hello world!

In [2]:
```

Permissions: **RW** End-of-lines: **LF** Encoding: **ASCII** Line: **1** Column: **1** Memory: **47 %**