

Linux Introduction

Bioinformatics Class 2015

- Presentation is a mixture of presentations that I found on the web from:
  - Εμμανουήλ Νίνος
  - Μιχάλης Ζήσης
  - Eric Bishop

# ΤΟ ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ **GNU/LINUX** **UBUNTU**

ΕΜΜΑΝΟΥΗΛ ΝΙΝΟΣ

Σχολική Βιβλιοθήκη

στο 9ο Γυμνάσιο Περιστερίου

You can't teach an old dog new tricks...



# Μερικοί μύθοι για το ελεύθερο λειτουργικό GNU/LINUX, BSD



- Είναι δύσκολο, τα καταφέρνουν μόνον προγραμματιστές
- Όλο εντολές γράφεις
- Δεν έχει γραφικό περιβάλλον
- Δε βρίσκω εφαρμογές
- Δε βρίσκω παιχνίδια
- Δεν υπάρχει υποστήριξη
- Δεν υπάρχουν οι κατάλληλοι οδηγοί για εκτυπωτές, κάμερες κτλ.
- Δε θα μπορώ να κάνω αυτά που έκανα με τον παλιό μου λειτουργικό
- Δεν μπορεί να είναι καλό αφού είναι δωρεάν

# Τι σημαίνει ελεύθερο λογισμικό;

0. να τρέχω το πρόγραμμα
1. να βλέπω και να αλλάζω τον κώδικα
2. να διανέμω αντίγραφα του προγράμματος
3. να διανέμω αντίγραφα του τροποποιημένου κώδικα

**Richard Stallman**

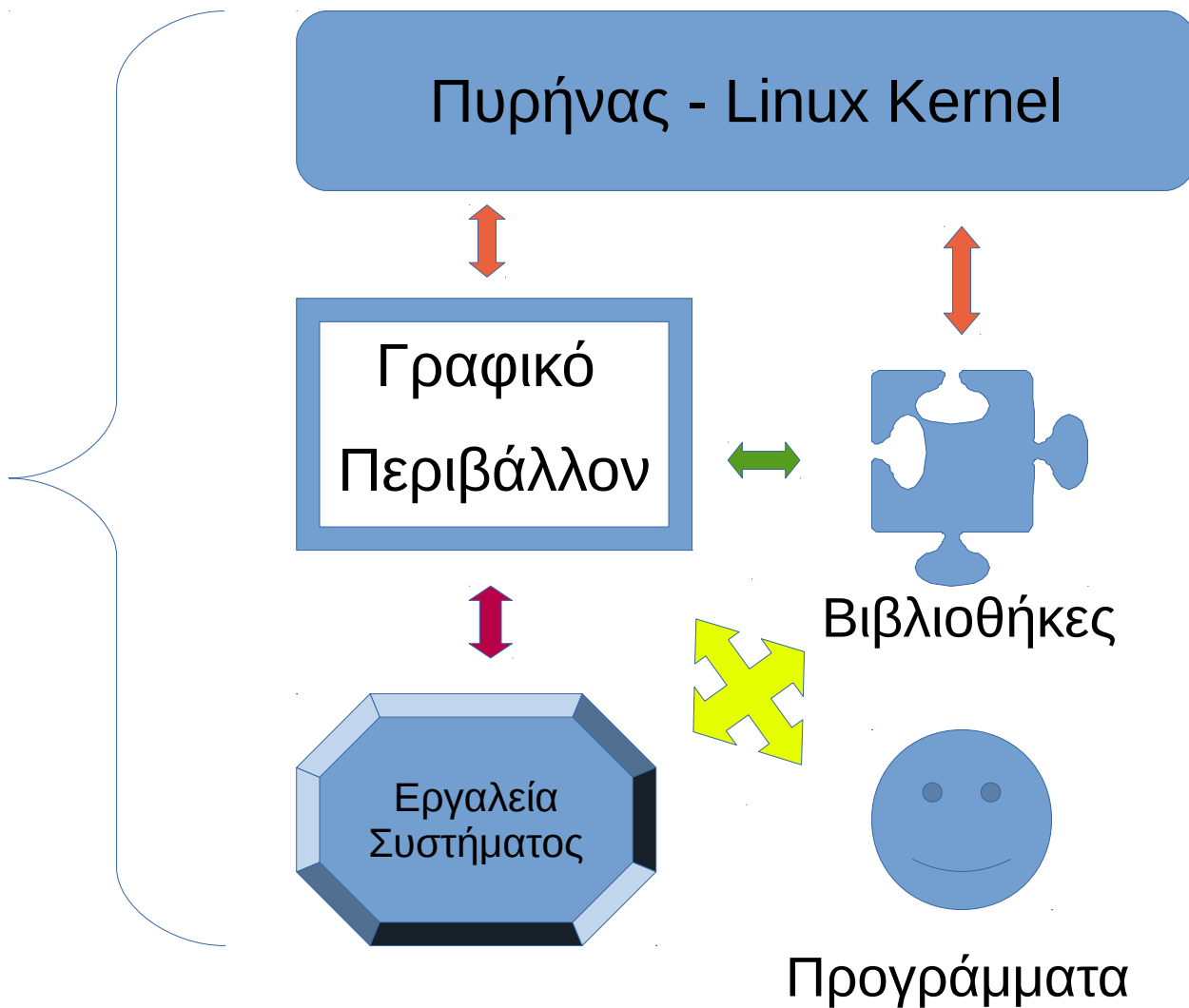
# Πόσες διανομές υπάρχουν;

Debian, Ubuntu, Fedora, openSUSE, Mint,  
Mandriva, Arch, Slackware, Puppy Linux,  
Gentoo, Ultimate Edition, Knoppix, Moblin,  
Xandros, Chrome OS, Sidux, Sabayon  
[sxolinux](#), [Monomaxos](#), [Zeus](#), [iloog](#),  
και δεκάδες άλλες,  
μερικές μάλιστα είναι ελληνικές!



# Διανομές GNU/Linux



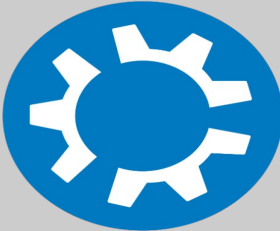


Διανομή = ...







# Ubuntu – Εκδόσεις II

Έκδοση	Ubuntu	Ubuntu Gnome	Kubuntu	Xubuntu	Lubuntu
Λογότυπο					
Γραφικό Περιβάλλον	Unity	Gnome 3	KDE	XFCE	LXDE
Επεξ. Μνήμη Γραφικά	Dual Core 2GB 3D 256MB		Dual Core 1GB 3D 1024x768	2 Ghz 512MB 800x600	1 Ghz 256MB 800x600
Δίσκος	10GB		10GB	5GB	5GB

ubuntu



# Εύκολο στη χρήση

- Ακόμη και για αρχάριους
- Περιέχει προεγκαταστημένα διάφορα προγράμματα: κειμενογράφο, φυλλομετρητή διαδικτύου, σημειώσεις Tomboy, αγγλικό λεξικό, Video player, επεξεργαστή εικόνας, παιχνίδια και πολλά άλλα!

Δείτε στις "εφαρμογές"

# Είναι πολύ ασφαλές

- Δεν πληρώνετε δεκάρα για αντιβιοτικά & τείχη προστασίας, ούτε επιβαρύνετε τον υπολογιστή σας καταναλώνοντας πολύτιμη υπολογιστική ισχύ.
- Οι ενημερώσεις ασφαλείας σας στέλνονται δωρεάν (δες διαχείριση ενημερώσεων).
- Τα νέα προγράμματα που προσθέτουμε με το Synaptic Package Manager, ή από το κέντρο λογισμικού Ubuntu είναι ελεγμένα από την Canonical & την κοινότητα.

# Είναι δωρεάν & νόμιμο

- Το κατεβάζετε από το διαδίκτυο, ή το εγκαθιστάτε από CD ή DVD, απολύτως νόμιμα

# Ταχύτατο!

- Διαχειρίζεται έξυπνα τους πόρους του συστήματος & σπανιότατα κολλάει.
- Σπανιότατα απαιτεί αυτόματο έλεγχο του δίσκου.

# Αξιόπιστο και σταθερό

- Όσα προγράμματα να προσθαιρέσετε δε "σέρνεται", ούτε χρειάζεται διαμόρφωση (format) του δίσκου.
- Απαιτείται επανεκκίνηση, μόνον όταν εγκαθιστάτε νέο πυρήνα Linux.

# Δουλεύει και σε παλιά μηχανήματα

Υπάρχουν μάλιστα πιο “ελαφριές” εκδόσεις που λειτουργούν ικανοποιητικότερα και με παλιότερα μηχανήματα.

Με άλλα λόγια, ο υπολογιστής οκταετίας που έχετε μπορεί να κάνει σχεδόν τα πάντα με την **τελευταία έκδοση του Ubuntu!** Ελέγξτε λοιπόν το σύστημά σας.



# Ευκολία εγκατάστασης & ενημέρωσης προγραμμάτων

Απλώς επιλέγεις μέσω του

διαχειριστή πακέτων,

ό,τι χρειάζεστε

και από εκεί και πέρα το ίδιο το σύστημα αναλαμβάνει

να τα κατεβάσει και να τα εγκαταστήσει.

**Το αποθετήριο έχει ελέγξει την ασφάλειά τους!**

# Υποστήριξη

Το μεγάλο πλεονέκτημα του Ubuntu.

Λαμβάνεις απάντηση στο ερώτημά σου, σχεδόν πάντα, αυθημερόν!

Τα ερωτήματα υποβάλλονται στα ελληνικά τόσο στο φόρουμ της ελληνικής κοινότητας, όσο και στο φόρουμ της διεθνούς.

# Πολλοί τρόποι χρήσης του λειτουργικού

- Από φλασάκι [live flash drive], χωρίς να εγκαταστήσεις τίποτε στο δίσκο
- Με διπλή εγκατάσταση [dual boot], κρατάς το παλιό σου λειτουργικό, και σε άλλη κατάτμηση του δίσκου έχεις το Ubuntu
- Από εικονική μηχανή
- Αφού δεις πόσο καλύτερο είναι το Ubuntu, απλά πετάς στα σκουπίδια το ελαττωματικό λειτουργικό που χρησιμοποιούσες παλιά και απολαμβάνεις ΜΟΝΟΝ το ελεύθερο λογισμικό

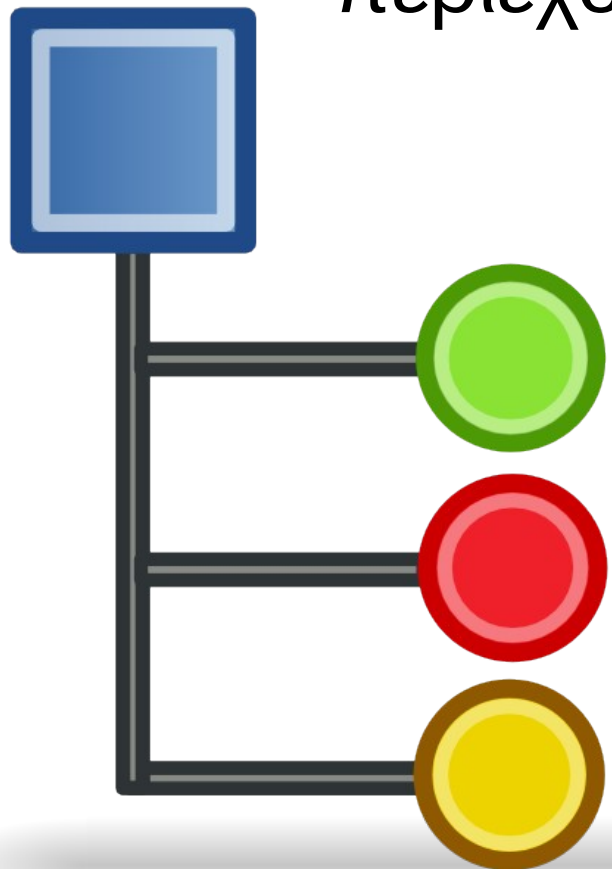
Μέσα στο λειτουργικό

- Η πιο συνηθισμένη ερώτηση είναι
- 
- 
- ΠΟΥ ΕΙΝΑΙ ΤΑ ΑΡΧΕΙΑ ΜΟΥ



# Σύστημα αρχείων III

Το Filesystem Hierarchy Standard (*FHS*) καθορίζει τη δομή του καταλόγου και τα περιεχόμενα καταλόγου σε Linux ΛΣ



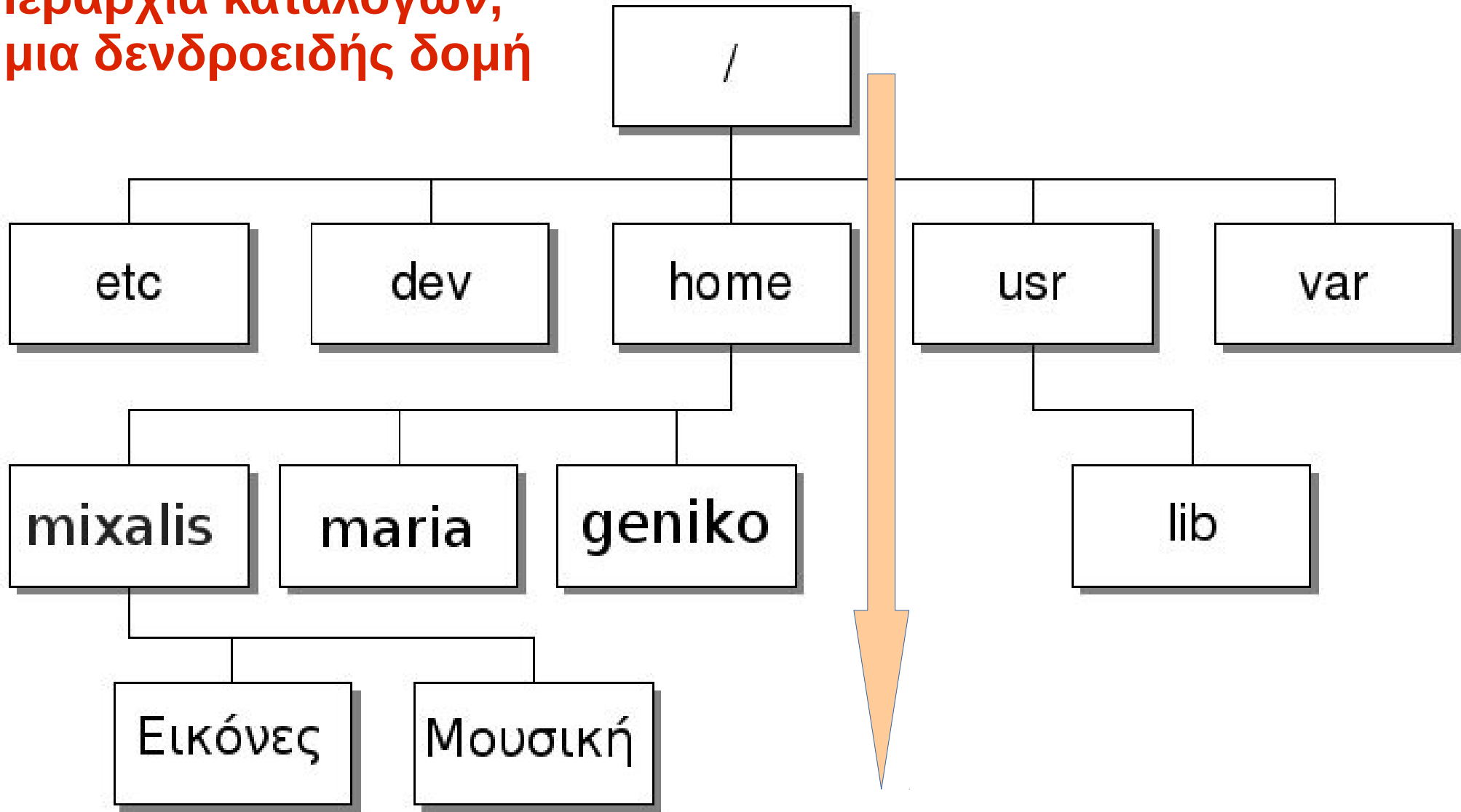
Όλα τα αρχεία και καταλόγοι εμφανίζονται κάτω από το ριζικό κατάλογο **"/"**

Πχ τα έγγραφα του χρήστη είναι στο: **`/home/xristis/Εικόνες`**



# Σύστημα αρχείων IV

Ιεραρχία καταλόγων,  
μια δενδροειδής δομή



# Why Linux?

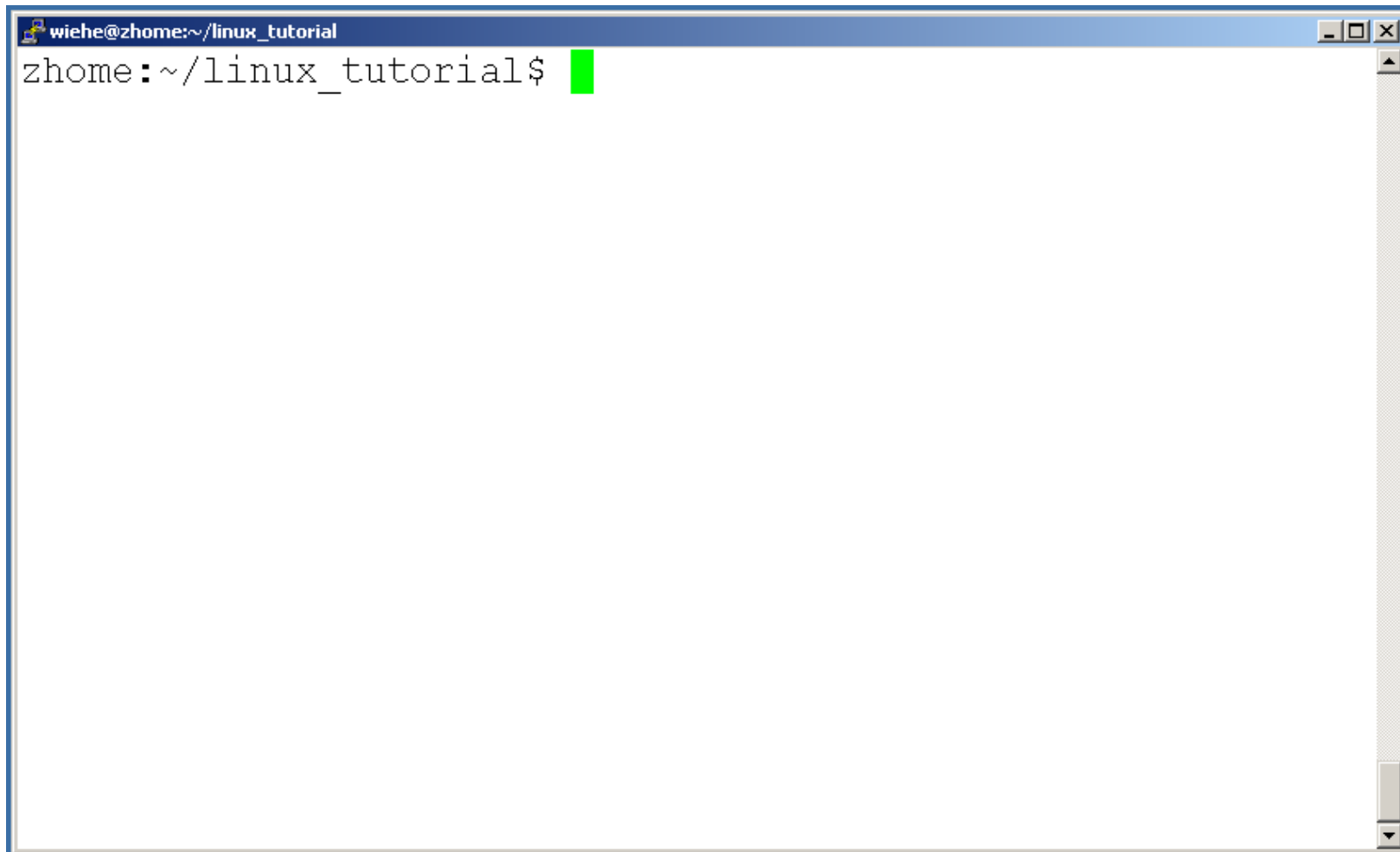
Powerful (remote) shell





# Connecting to a Unix/Linux system

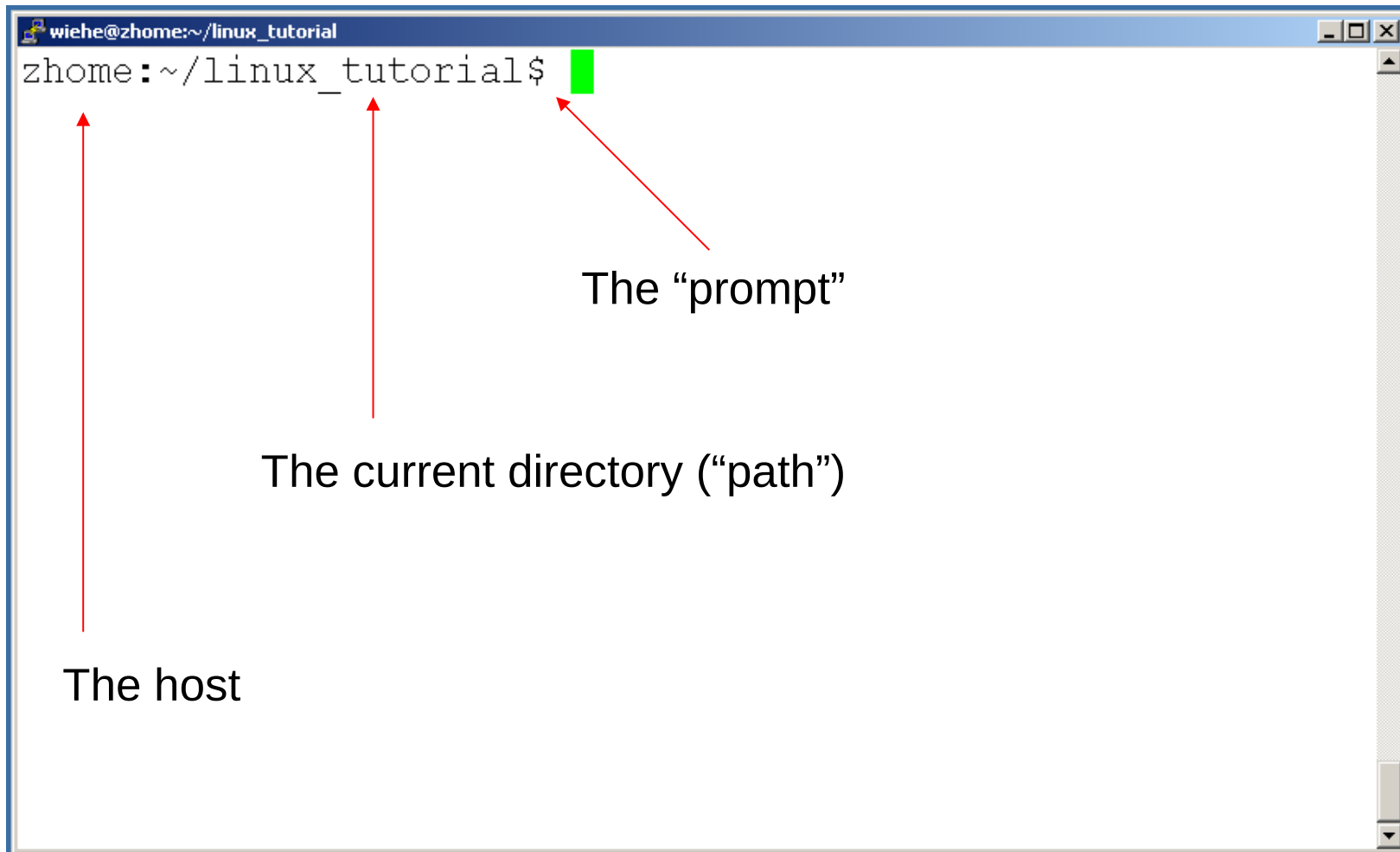
- Open up a terminal:

A screenshot of a terminal window. The title bar at the top reads "wiehe@zhome:~/linux\_tutorial" and includes standard window control buttons (minimize, maximize, close). The terminal content shows the prompt "zhome:~/linux\_tutorial\$" followed by a green cursor block. The rest of the terminal area is empty.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ █
```

# Connecting to a Unix/Linux system

- Open up a terminal:



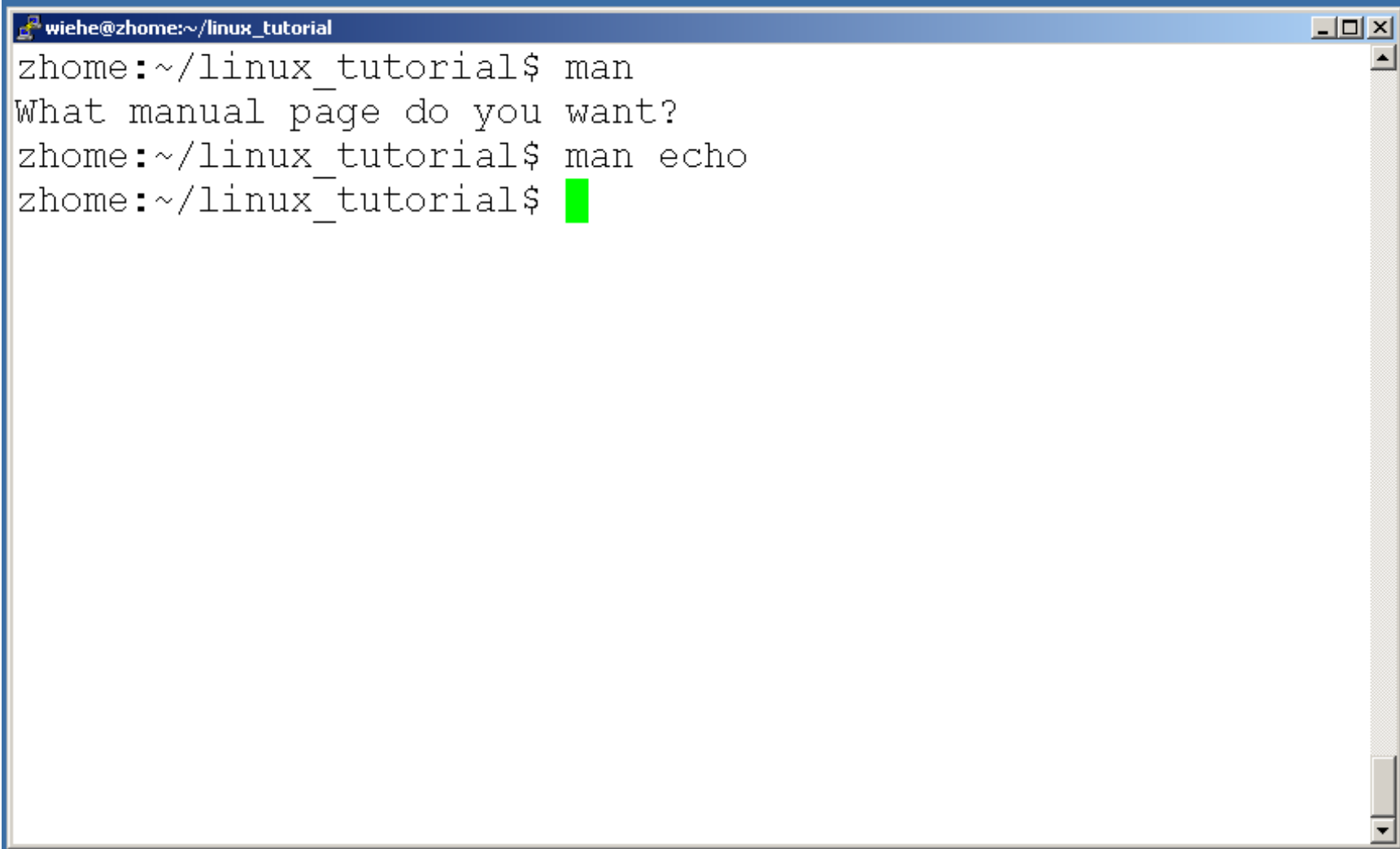
# What exactly is a “shell”?

- After logging in, Linux/Unix starts another program called the **shell**
- The shell interprets commands the user types and manages their execution
  - The shell communicates with the internal part of the operating system called the **kernel**
  - The most popular shells are: tcsh, csh, korn, and bash
  - The differences are most times subtle
  - For this tutorial, we are using bash
- Shell commands are **CASE SENSITIVE!**

# Help!

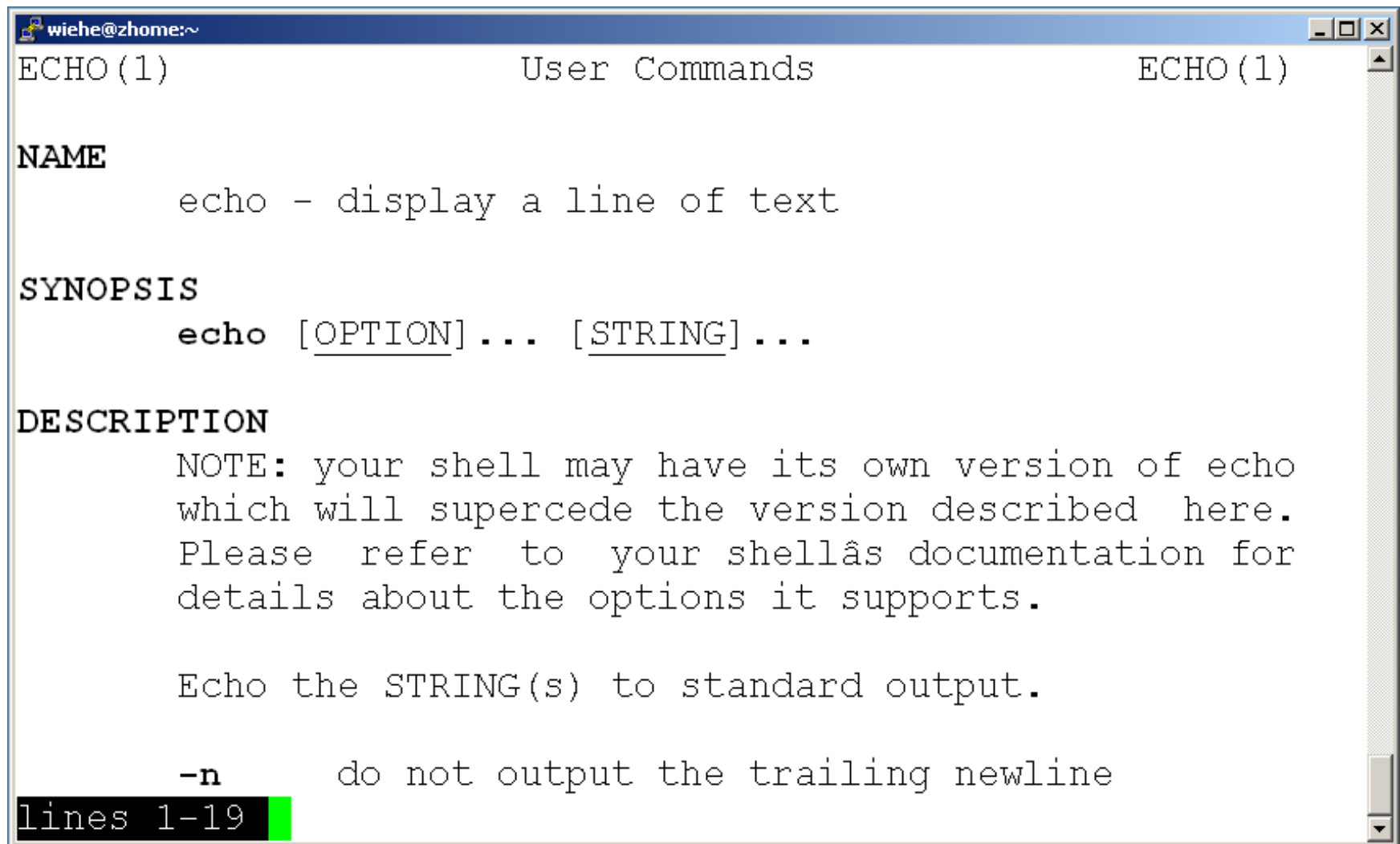
- Whenever you need help with a command type “man” and the command name

# Help!



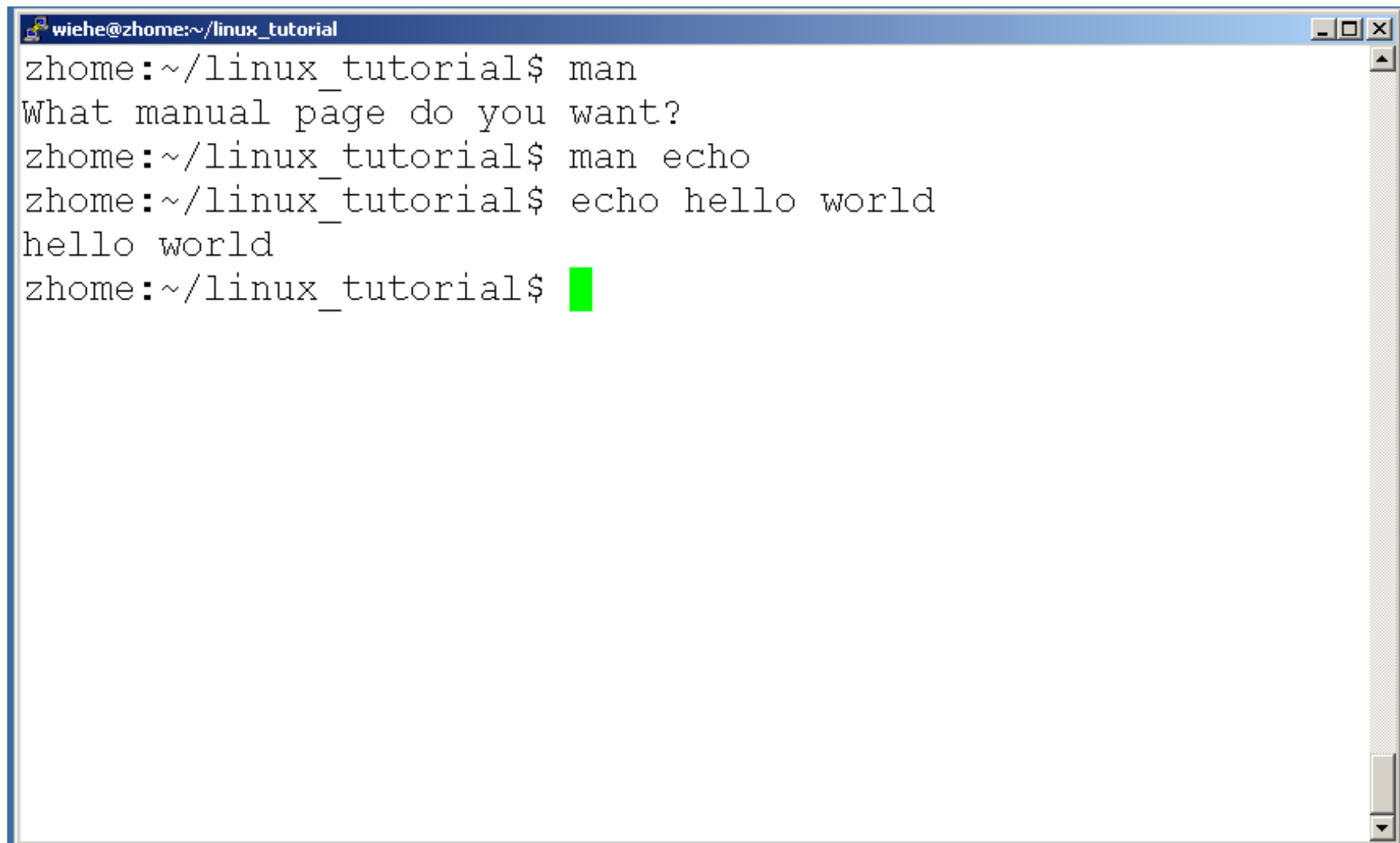
```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ man
What manual page do you want?
zhome:~/linux_tutorial$ man echo
zhome:~/linux_tutorial$ █
```

# Help!



```
wiehe@zhome:~  
ECHO (1) User Commands ECHO (1)  
  
NAME  
    echo - display a line of text  
  
SYNOPSIS  
    echo [OPTION]... [STRING]...  
  
DESCRIPTION  
    NOTE: your shell may have its own version of echo  
    which will supercede the version described here.  
    Please refer to your shell's documentation for  
    details about the options it supports.  
  
    Echo the STRING(s) to standard output.  
  
    -n    do not output the trailing newline  
lines 1-19
```

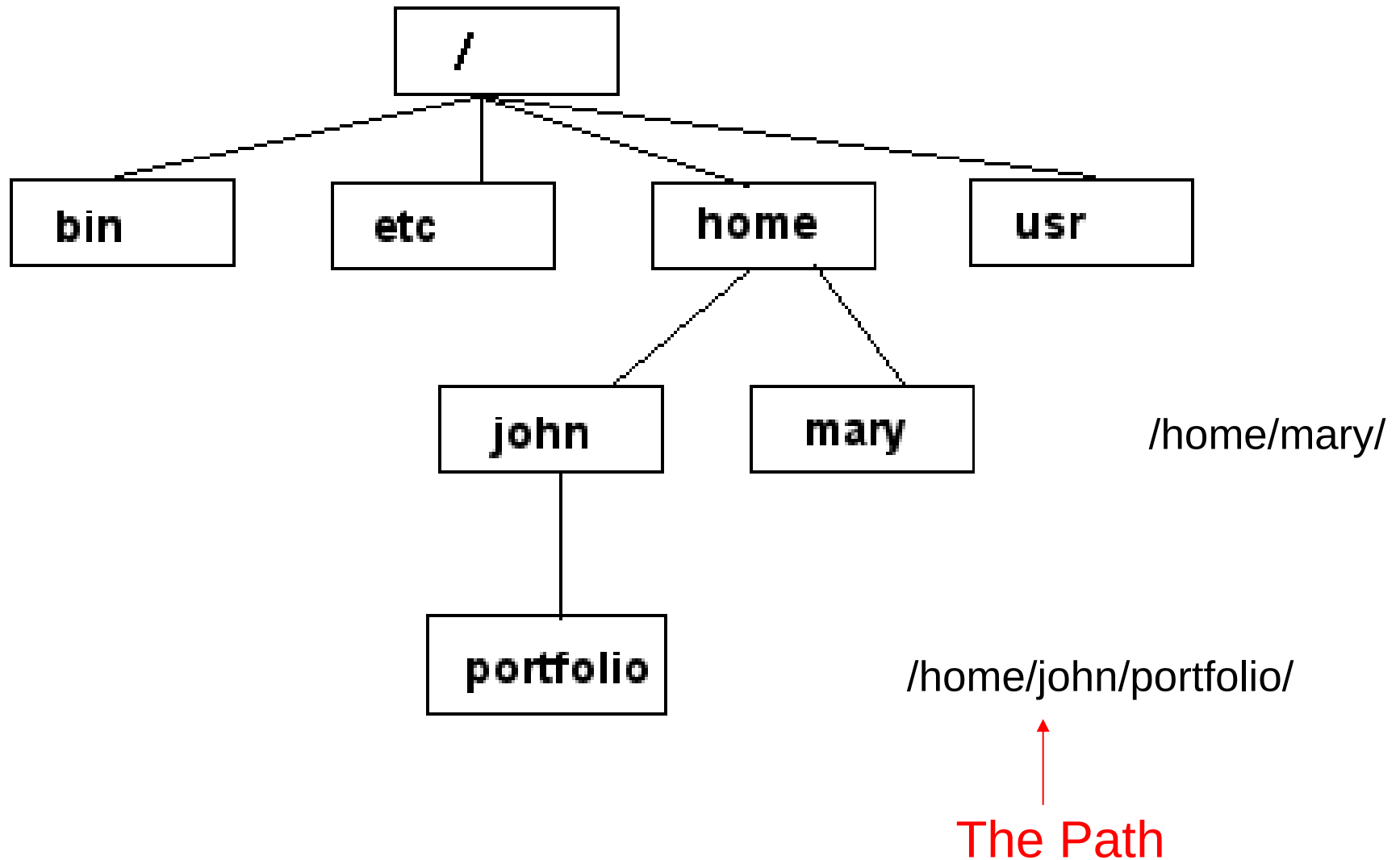
# Help!



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ man
What manual page do you want?
zhome:~/linux_tutorial$ man echo
zhome:~/linux_tutorial$ echo hello world
hello world
zhome:~/linux_tutorial$ █
```

# Unix/Linux File System

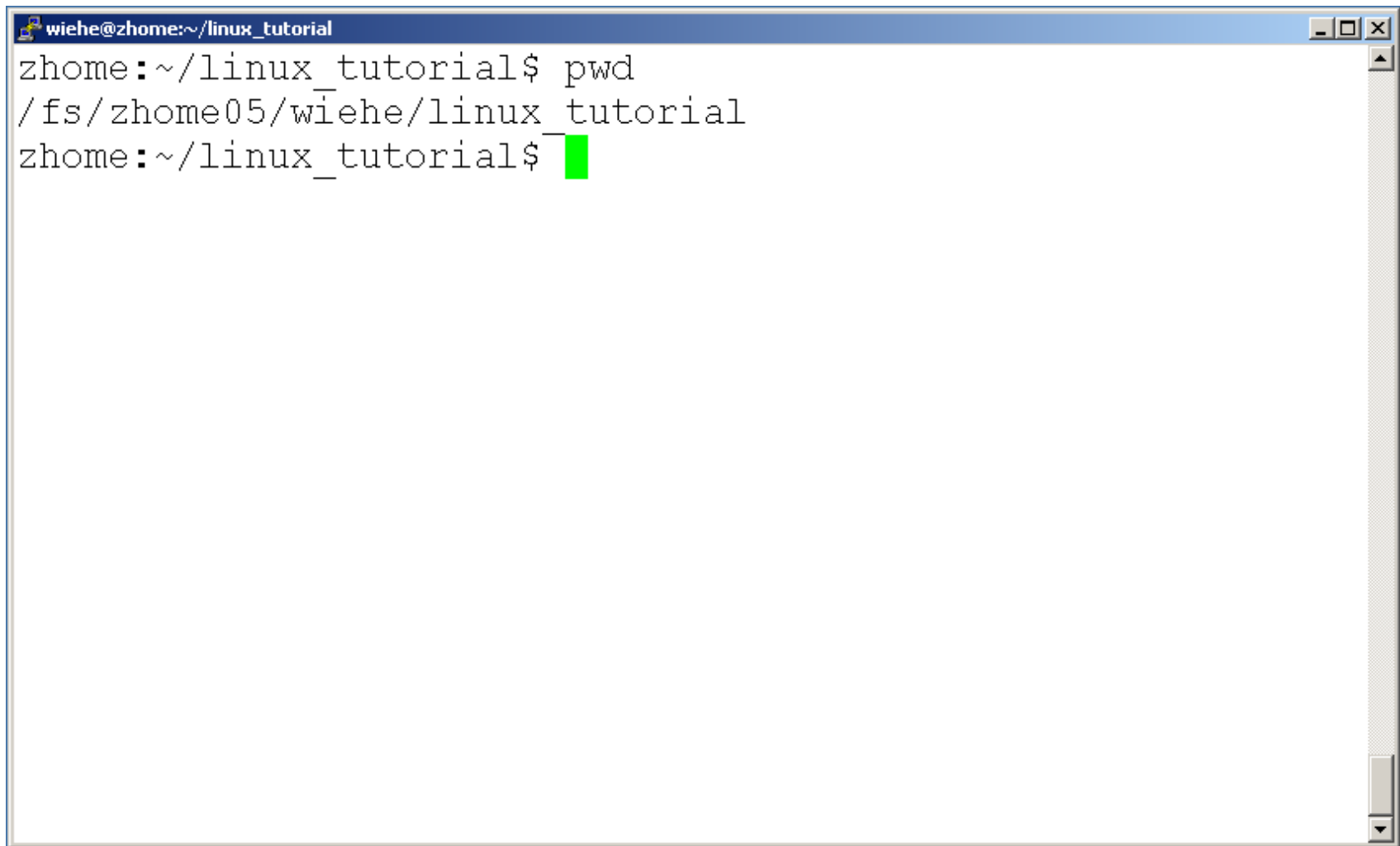
NOTE: Unix file names are **CASE SENSITIVE!**





# Command: pwd

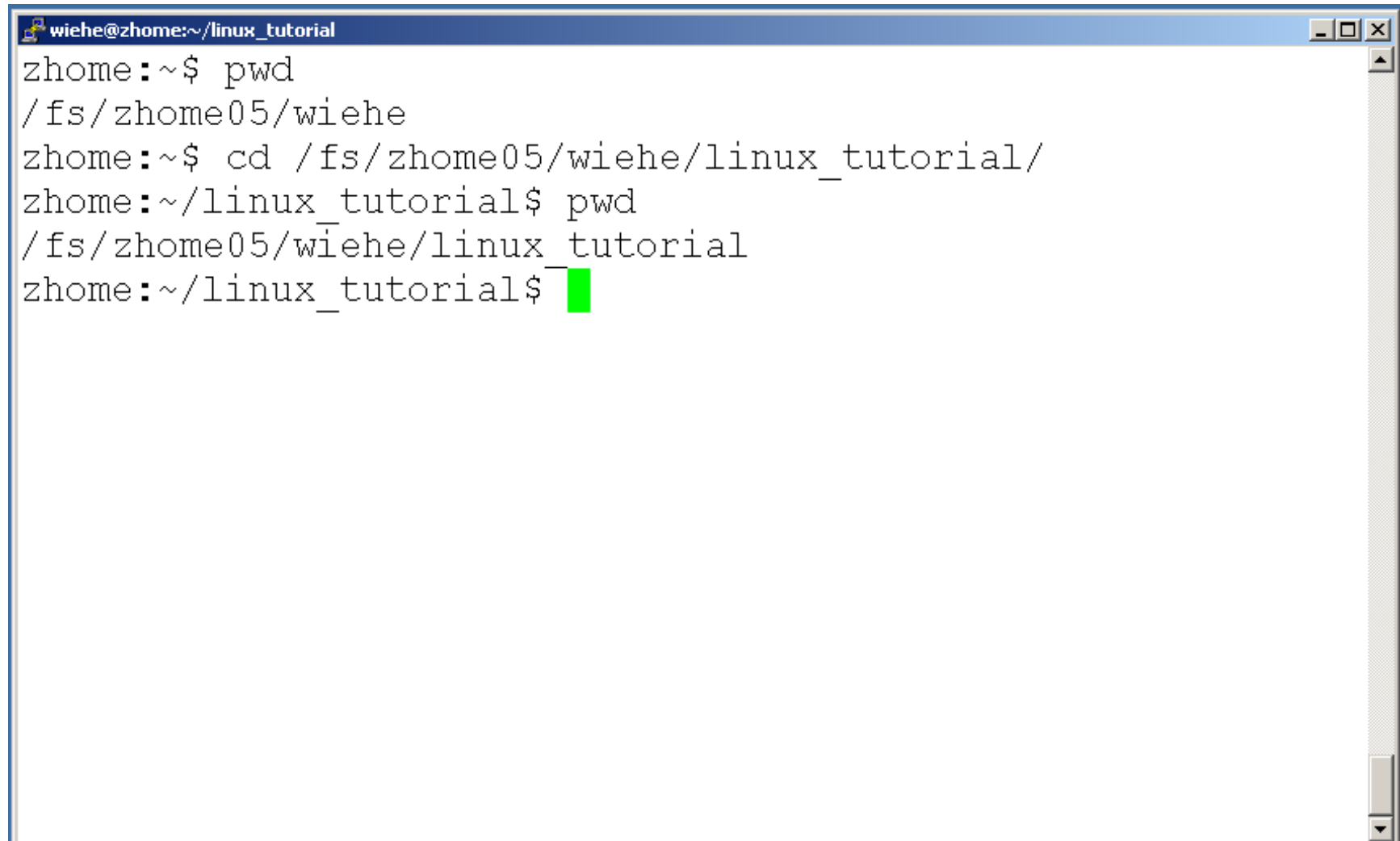
- To find your current path use “pwd”

A terminal window with a blue title bar containing the text "wiehe@zhome:~/linux\_tutorial". The terminal content shows the command "pwd" being executed, resulting in the output "/fs/zhome05/wiehe/linux\_tutorial". The prompt "zhome:~/linux\_tutorial\$" is shown before and after the command. A green cursor is visible at the end of the second prompt.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ pwd
/fs/zhome05/wiehe/linux_tutorial
zhome:~/linux_tutorial$
```

# Command: cd

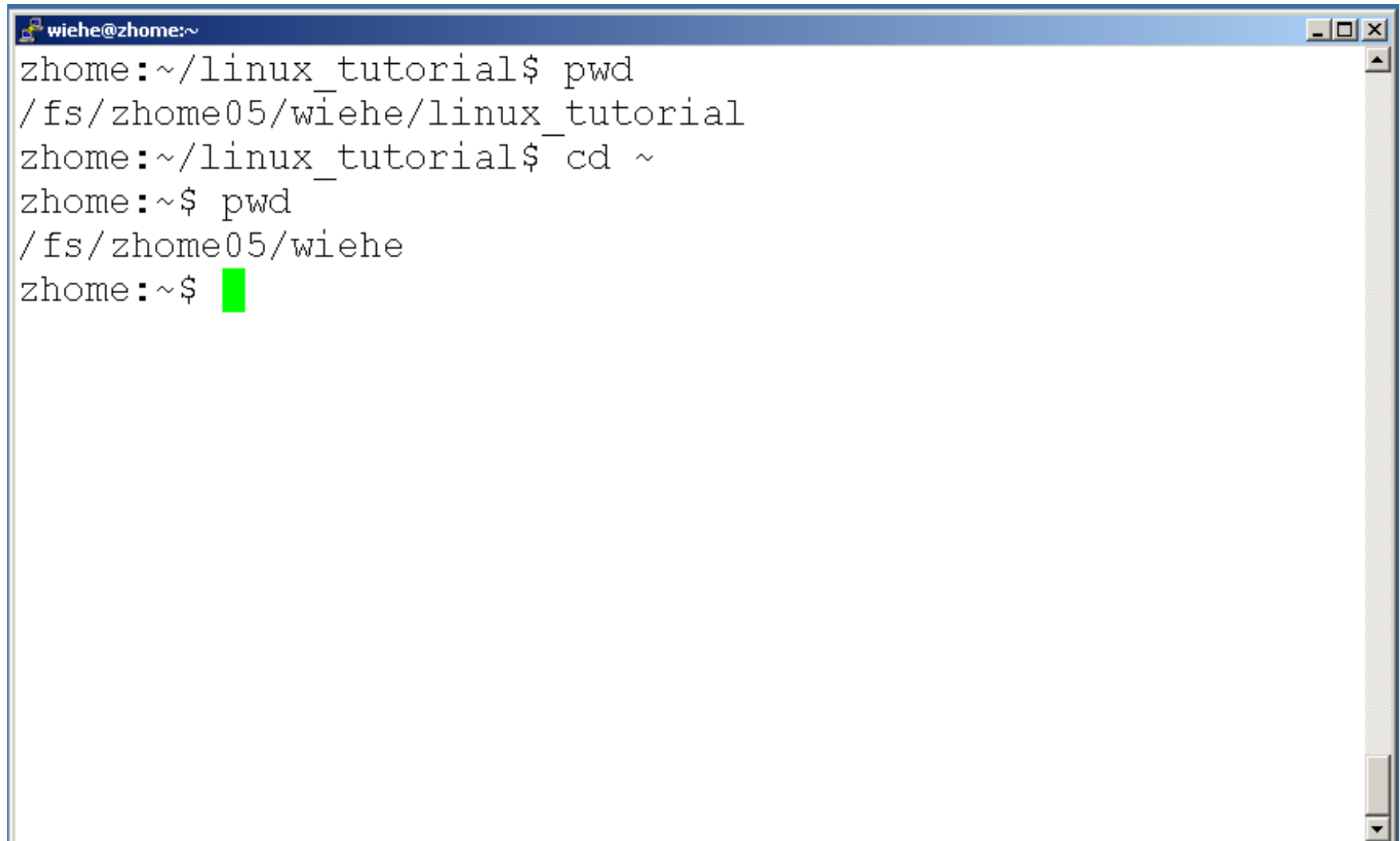
- To change to a specific directory use “cd”



```
wiehe@zhome:~/linux_tutorial
zhome:~$ pwd
/fs/zhome05/wiehe
zhome:~$ cd /fs/zhome05/wiehe/linux_tutorial/
zhome:~/linux_tutorial$ pwd
/fs/zhome05/wiehe/linux_tutorial
zhome:~/linux_tutorial$ █
```

# Command: cd

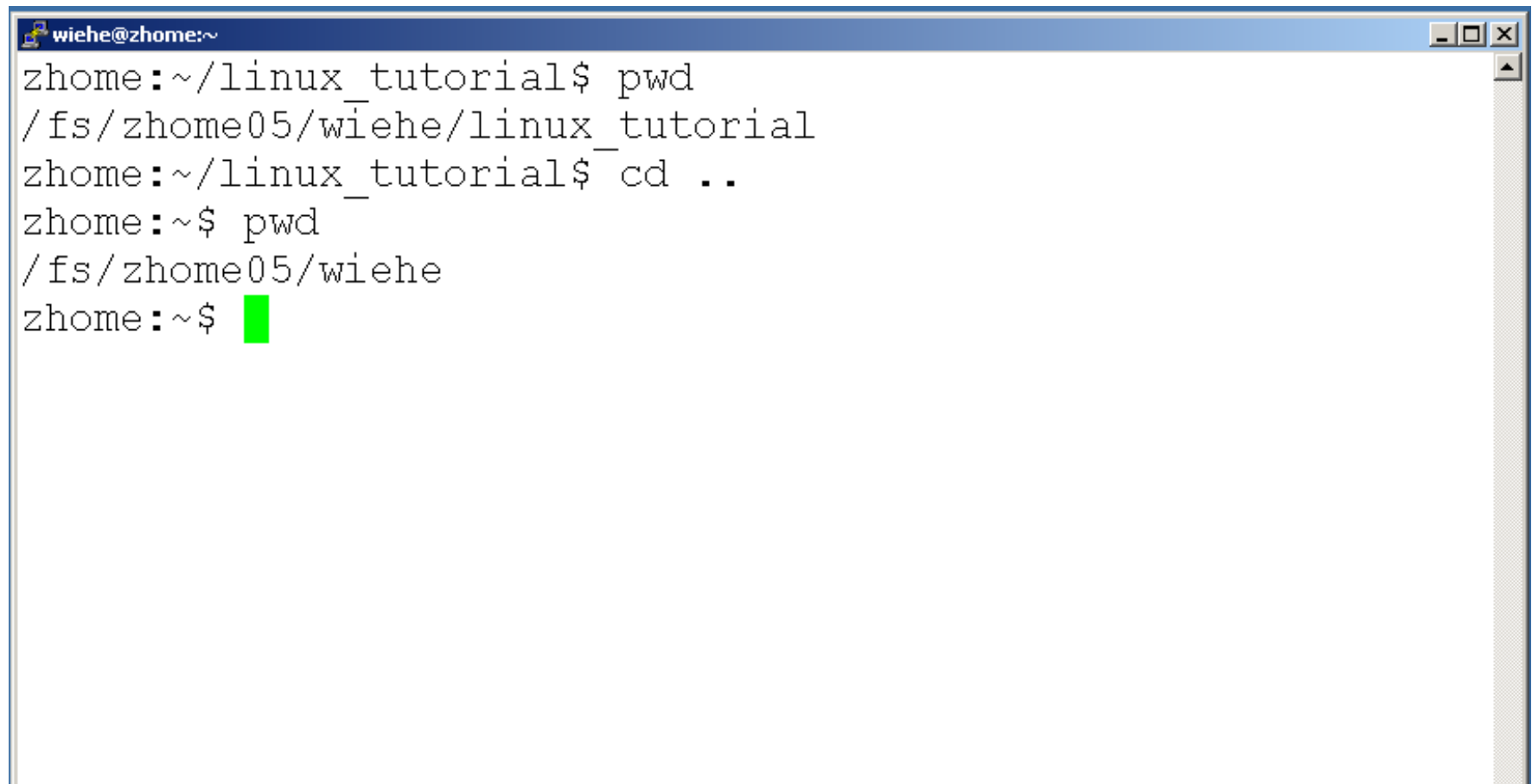
- “~” is the location of your home directory



```
wiehe@zhome:~  
zhome:~/linux_tutorial$ pwd  
/fs/zhome05/wiehe/linux_tutorial  
zhome:~/linux_tutorial$ cd ~  
zhome:~$ pwd  
/fs/zhome05/wiehe  
zhome:~$ █
```

# Command: cd

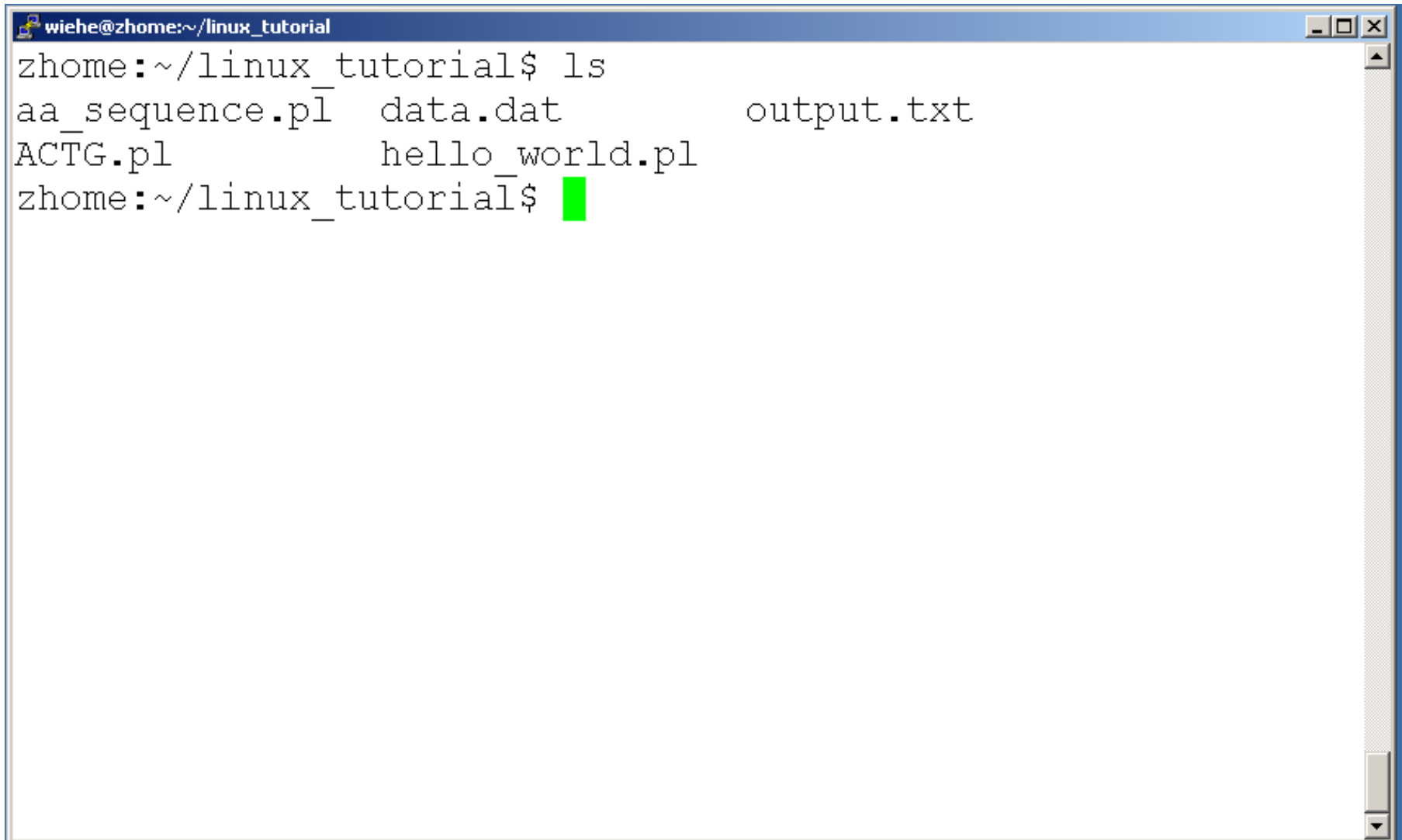
- “..” is the location of the directory below current one

A terminal window titled 'wiehe@zhome:~' showing a sequence of commands and their outputs. The user starts in the directory ~/linux\_tutorial, runs 'pwd' to get /fs/zhome05/wiehe/linux\_tutorial, then runs 'cd ..' to move to the parent directory. Finally, they run 'pwd' again to get /fs/zhome05/wiehe. A green cursor is visible at the end of the final prompt.

```
wiehe@zhome:~  
zhome:~/linux_tutorial$ pwd  
/fs/zhome05/wiehe/linux_tutorial  
zhome:~/linux_tutorial$ cd ..  
zhome:~$ pwd  
/fs/zhome05/wiehe  
zhome:~$ █
```

# Command: ls

- To list the files in the current directory use “ls”

A terminal window with a blue title bar containing the text 'wiehe@zhome:~/linux\_tutorial'. The terminal content shows the command 'ls' being executed, resulting in a listing of files: 'aa\_sequence.pl', 'data.dat', 'output.txt', and 'hello\_world.pl'. The prompt 'zhome:~/linux\_tutorial\$' is shown again at the end of the output, followed by a green cursor.

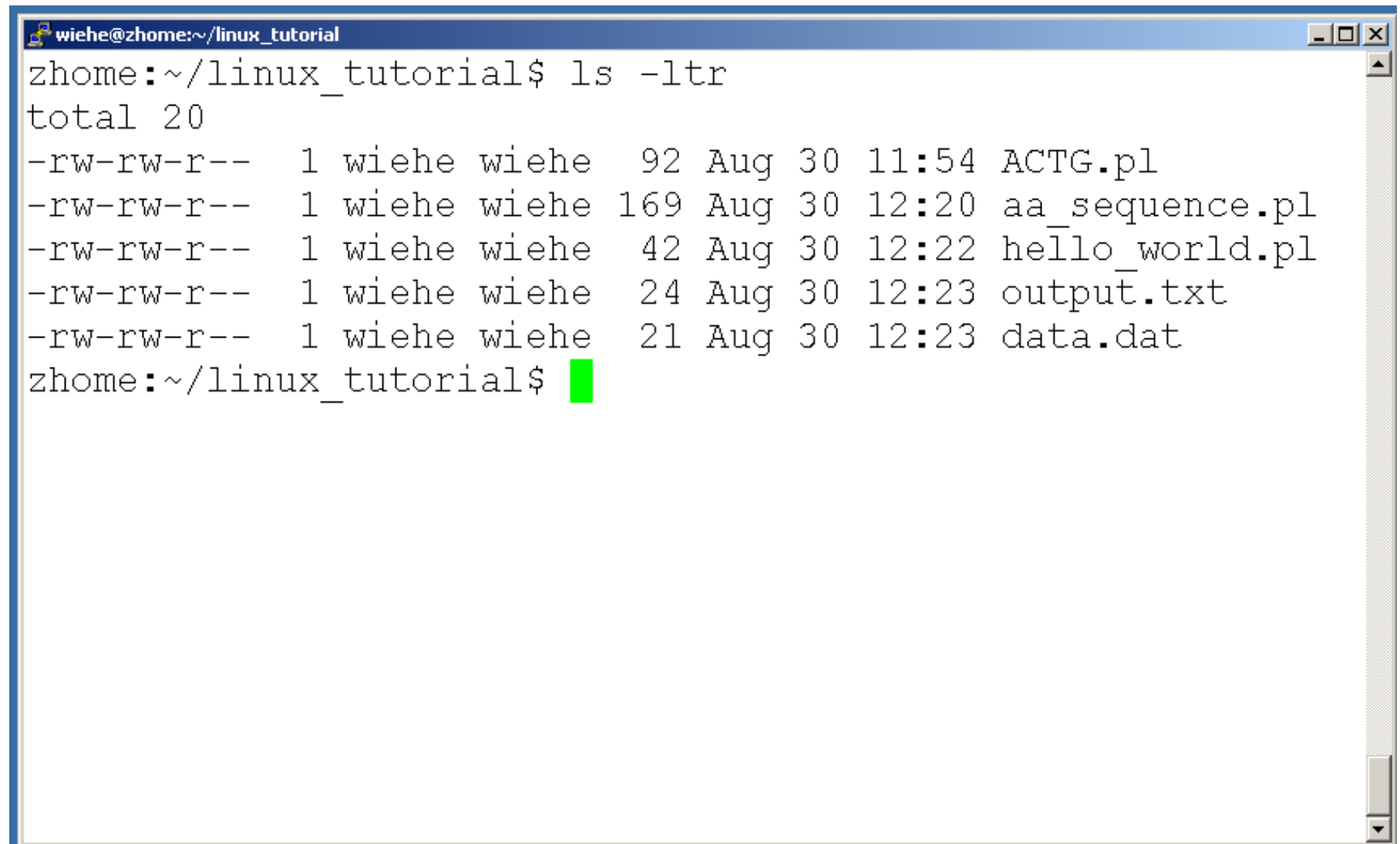
```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          output.txt
ACTG.pl        hello_world.pl
zhome:~/linux_tutorial$ █
```

# Command: ls

- ls has many options
  - -l long list (displays lots of info)
  - -t sort by modification time
  - -S sort by size
  - -h list file sizes in human readable format
  - -r reverse the order
- “man ls” for more options
- Options can be combined: “ls -ltr”

# Command: ls -ltr

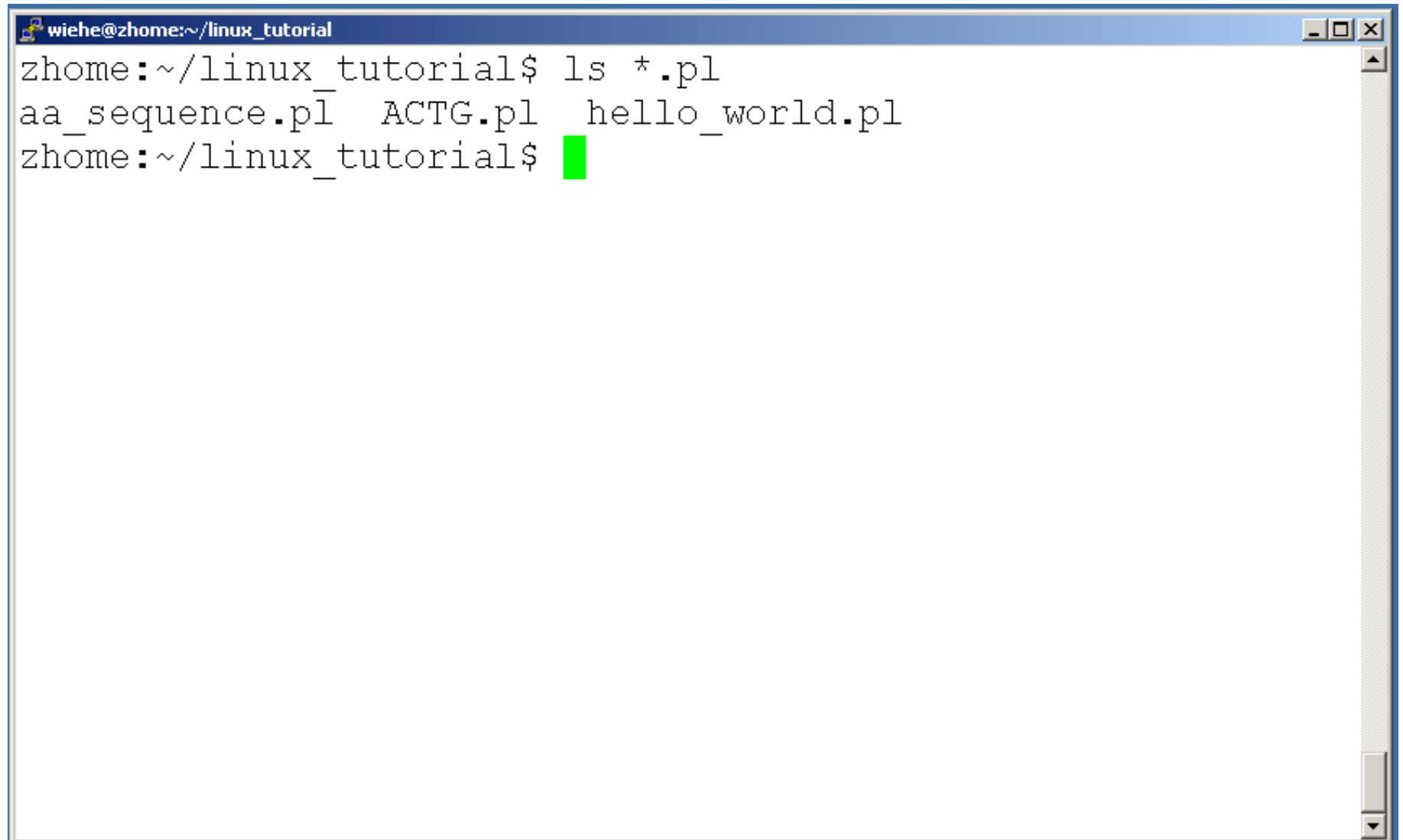
- List files by time in reverse order with long listing



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls -ltr
total 20
-rw-rw-r--  1 wiehe wiehe  92 Aug 30 11:54 ACTG.pl
-rw-rw-r--  1 wiehe wiehe 169 Aug 30 12:20 aa_sequence.pl
-rw-rw-r--  1 wiehe wiehe  42 Aug 30 12:22 hello_world.pl
-rw-rw-r--  1 wiehe wiehe  24 Aug 30 12:23 output.txt
-rw-rw-r--  1 wiehe wiehe  21 Aug 30 12:23 data.dat
zhome:~/linux_tutorial$ █
```

# General Syntax: \*

- “\*” can be used as a wildcard in unix/linux

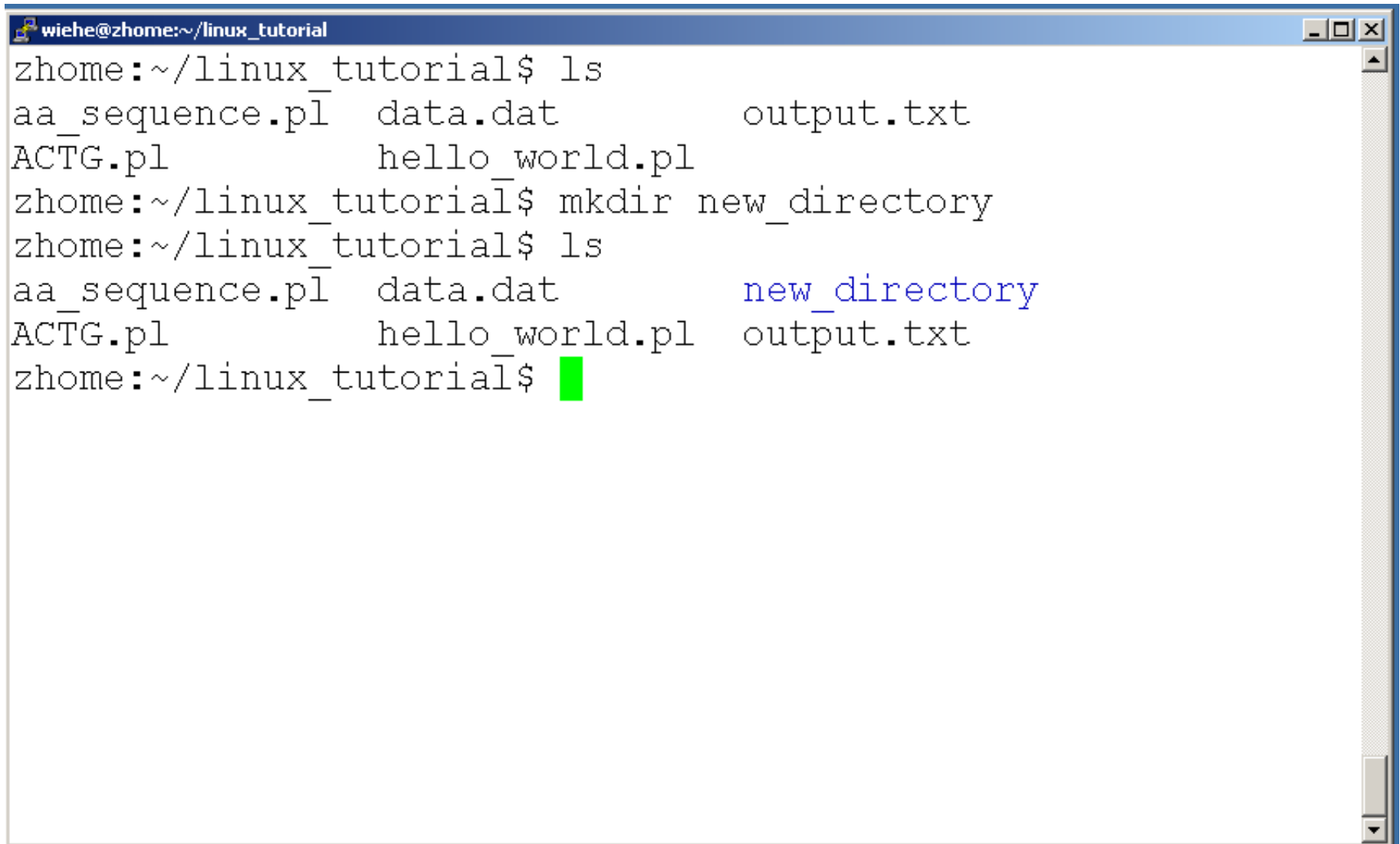
A terminal window with a blue title bar containing the text "wiehe@zhome:~/linux\_tutorial". The terminal content shows a command prompt "zhome:~/linux\_tutorial\$" followed by the command "ls \*.pl". The output of the command is "aa\_sequence.pl ACTG.pl hello\_world.pl". Below the output, the prompt "zhome:~/linux\_tutorial\$" is shown again with a green cursor block.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls *.pl
aa_sequence.pl  ACTG.pl  hello_world.pl
zhome:~/linux_tutorial$
```



# Command: mkdir

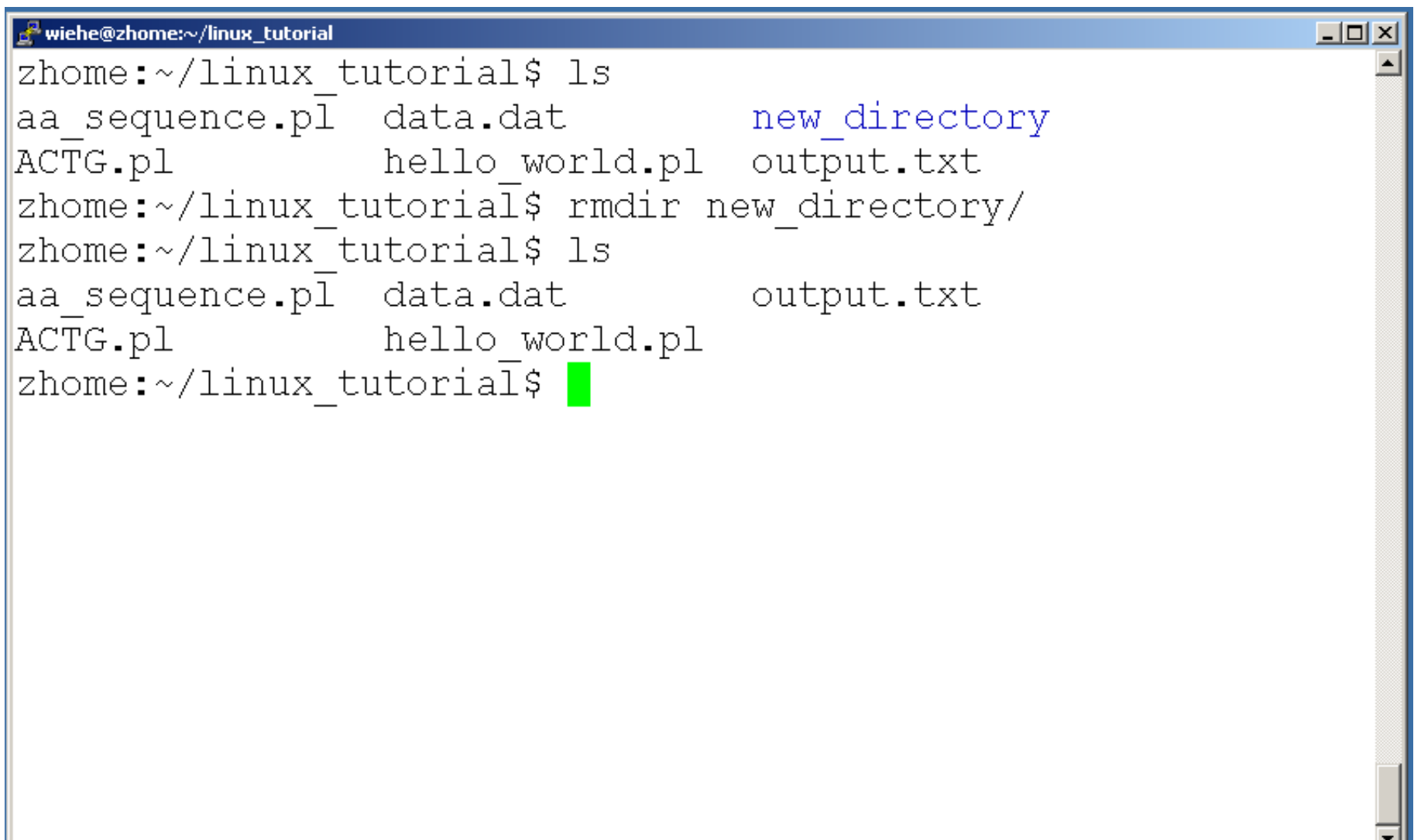
- To create a new directory use “mkdir”



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          output.txt
ACTG.pl        hello_world.pl
zhome:~/linux_tutorial$ mkdir new_directory
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          new_directory
ACTG.pl        hello_world.pl   output.txt
zhome:~/linux_tutorial$ █
```

# Command: rmdir

- To remove an empty directory use “rmdir”



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          new_directory
ACTG.pl        hello_world.pl   output.txt
zhome:~/linux_tutorial$ rmdir new_directory/
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          output.txt
ACTG.pl        hello_world.pl
zhome:~/linux_tutorial$ █
```

# Creating files in Unix/Linux

- Requires the use of an Editor
- Various Editors:
  - 1) nano / pico
  - 2) vi
  - 3) emacs



# Editing a file using pico or nano

- Type “pico” or “nano” at the prompt



The screenshot shows a terminal window with the title bar "wiehe@zhome:~/linux\_tutorial". The terminal content is as follows:

```
UW PICO (tm) 4.6      New Buffer
```

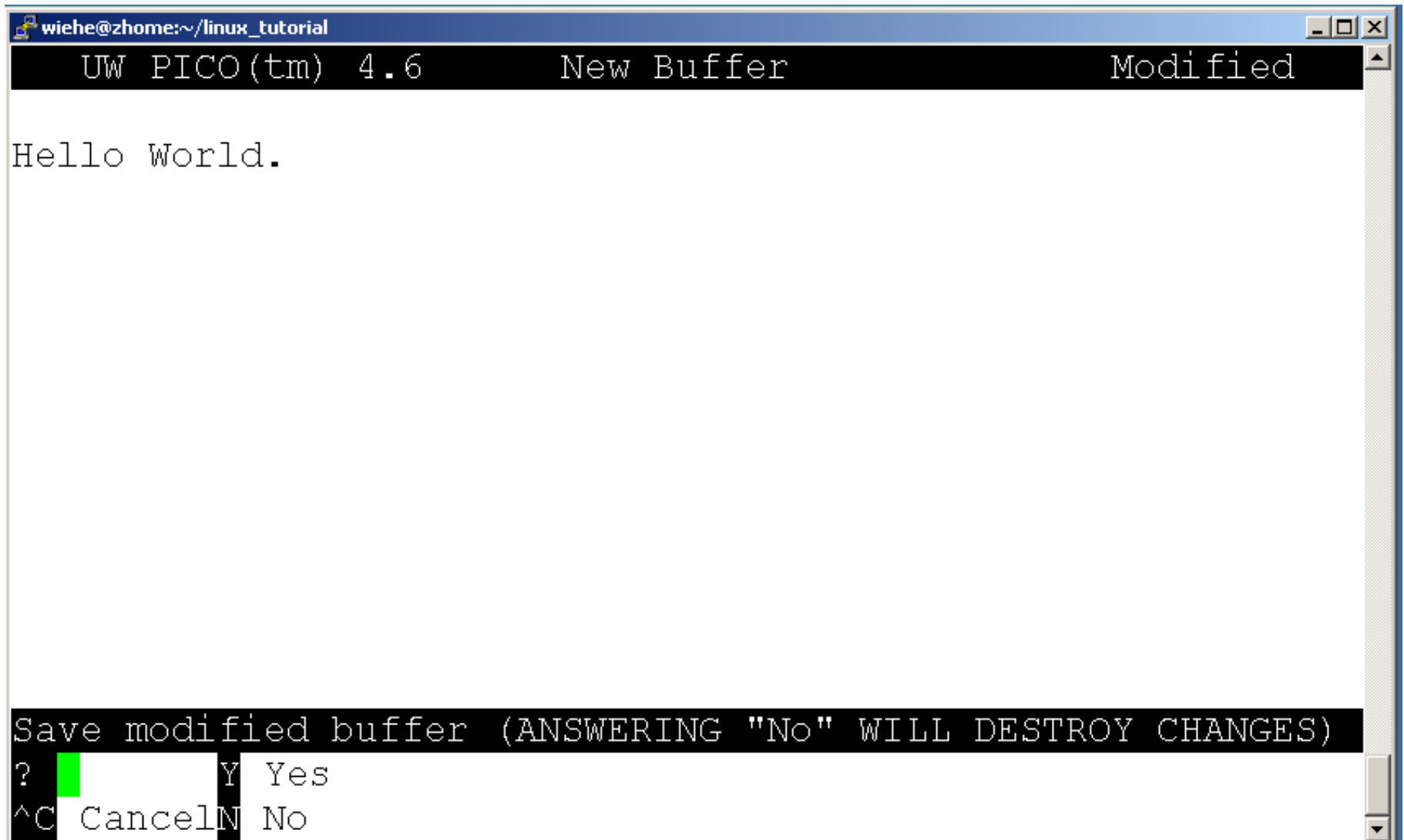
A green cursor is visible on the left side of the first line. On the right side of the terminal, there is a small box containing the text "Learning UNIX".

At the bottom of the terminal, a help menu is displayed:

```
^G Get He ^O WriteO ^R Read F ^Y Prev P ^K Cut Te ^C Cur Po  
^X Exit  ^J Justif ^W Where ^V Next P ^U UnCut ^T To Spe
```

# Editing a file using pico

- To save use “ctrl-x”



```
wiehe@zhome:~/linux_tutorial
UW PICO(tm) 4.6      New Buffer      Modified
Hello World.

Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES)
?  Yes
^C  No
```

# Displaying a file

- Various ways to display a file in Unix
  - cat
  - less
  - head
  - tail

# Command: cat

- Dumps an entire file to standard output
- Good for displaying short, simple files

# Command: less

- “less” displays a file, allowing forward/backward movement within it
  - return scrolls forward one line, space one page
  - y scrolls back one line, b one page
- use “/” to search for a string
- Press q to quit

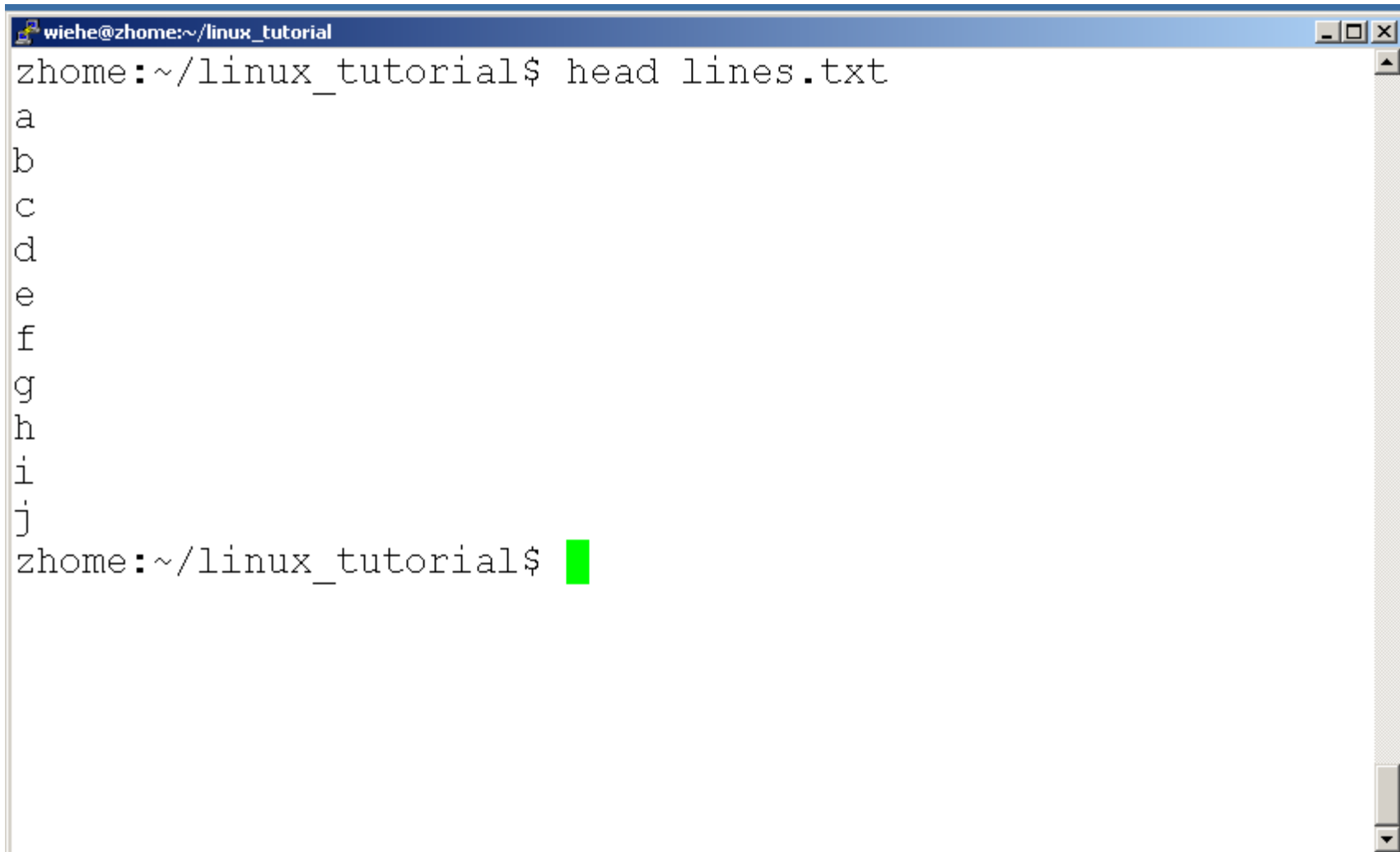


# Command: head

- “head” displays the top part of a file
- By default it shows the first 10 lines
- -n option allows you to change that
- “head -n50 file.txt” displays the first 50 lines of file.txt

# Command: head

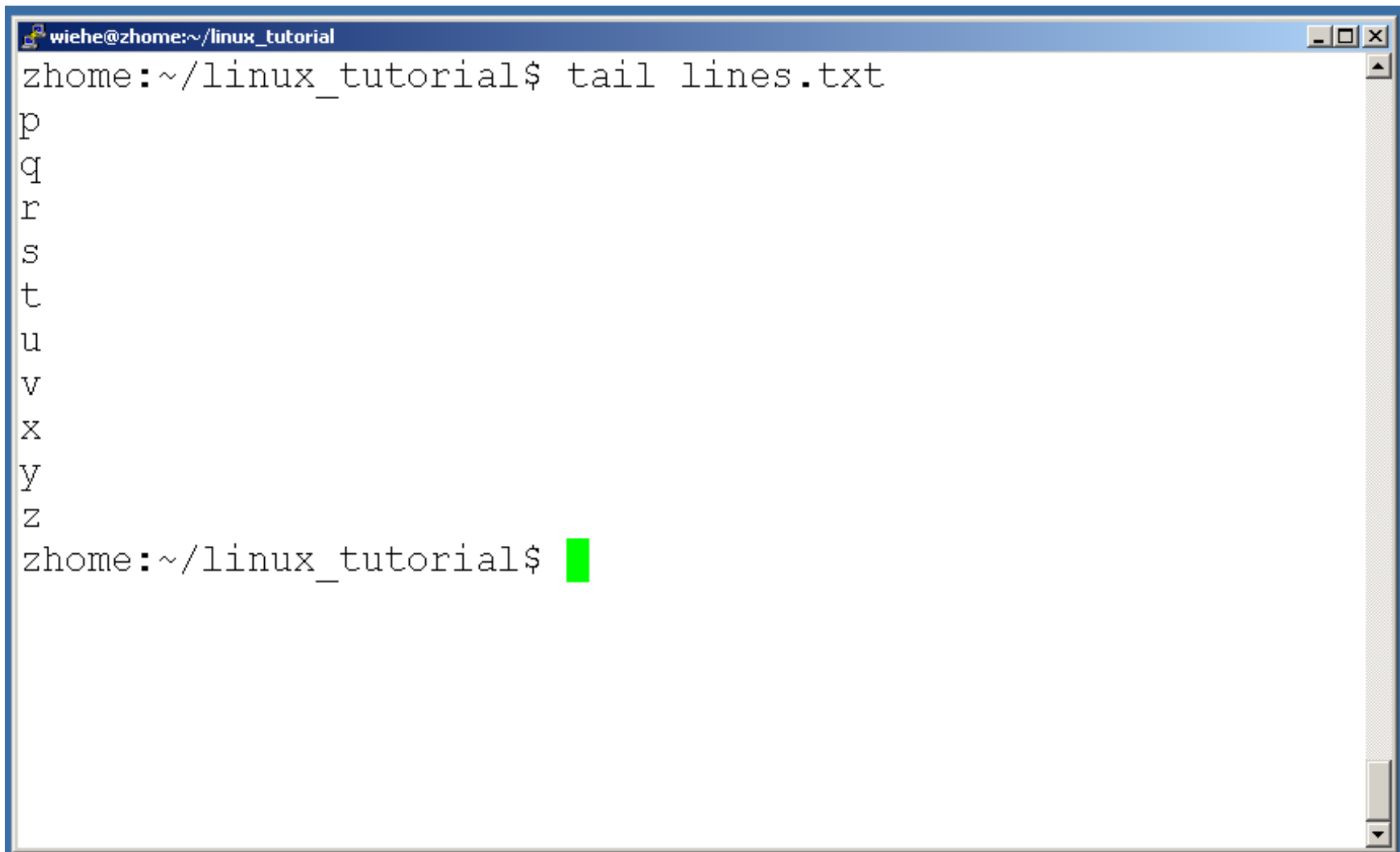
- Here's an example of using "head":

A terminal window with a blue title bar containing the text 'wiehe@zhome:~/linux\_tutorial'. The terminal shows the command 'head lines.txt' being executed. The output consists of ten lines of lowercase letters from 'a' to 'j'. The prompt 'zhome:~/linux\_tutorial\$' is visible at the end of the output, followed by a green cursor block.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ head lines.txt
a
b
c
d
e
f
g
h
i
j
zhome:~/linux_tutorial$ █
```

# Command: tail

- Same as head, but shows the last lines

A terminal window with a blue title bar containing the text 'wiehe@zhome:~/linux\_tutorial'. The terminal shows the command 'tail lines.txt' being executed, followed by the output 'p', 'q', 'r', 's', 't', 'u', 'v', 'x', 'y', 'z'. The prompt 'zhome:~/linux\_tutorial\$' is followed by a green cursor block.

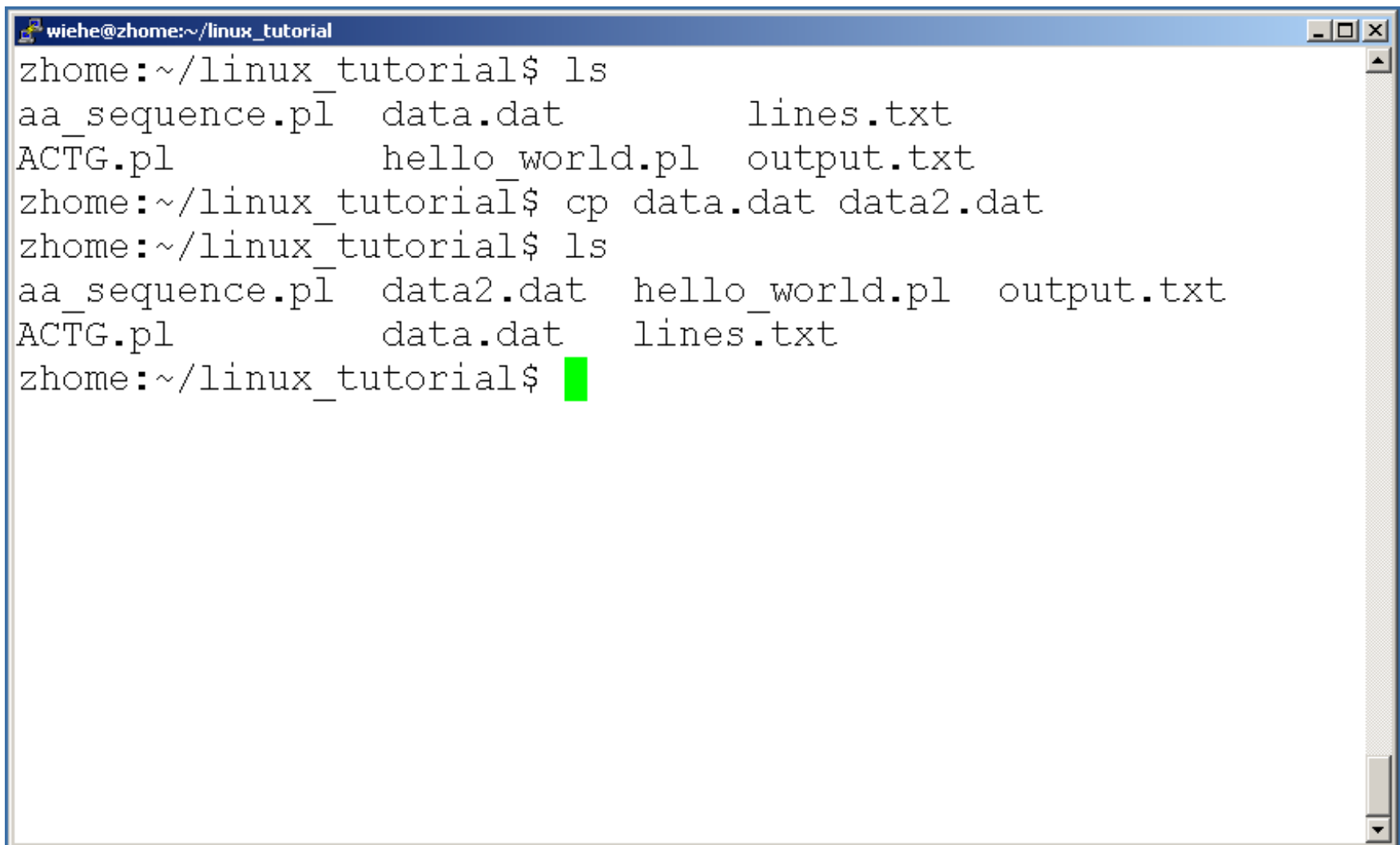
```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ tail lines.txt
p
q
r
s
t
u
v
x
y
z
zhome:~/linux_tutorial$ █
```

# File Commands

- Copying a file: `cp`
- Move or rename a file: `mv`
- Remove a file: `rm`

# Command: cp

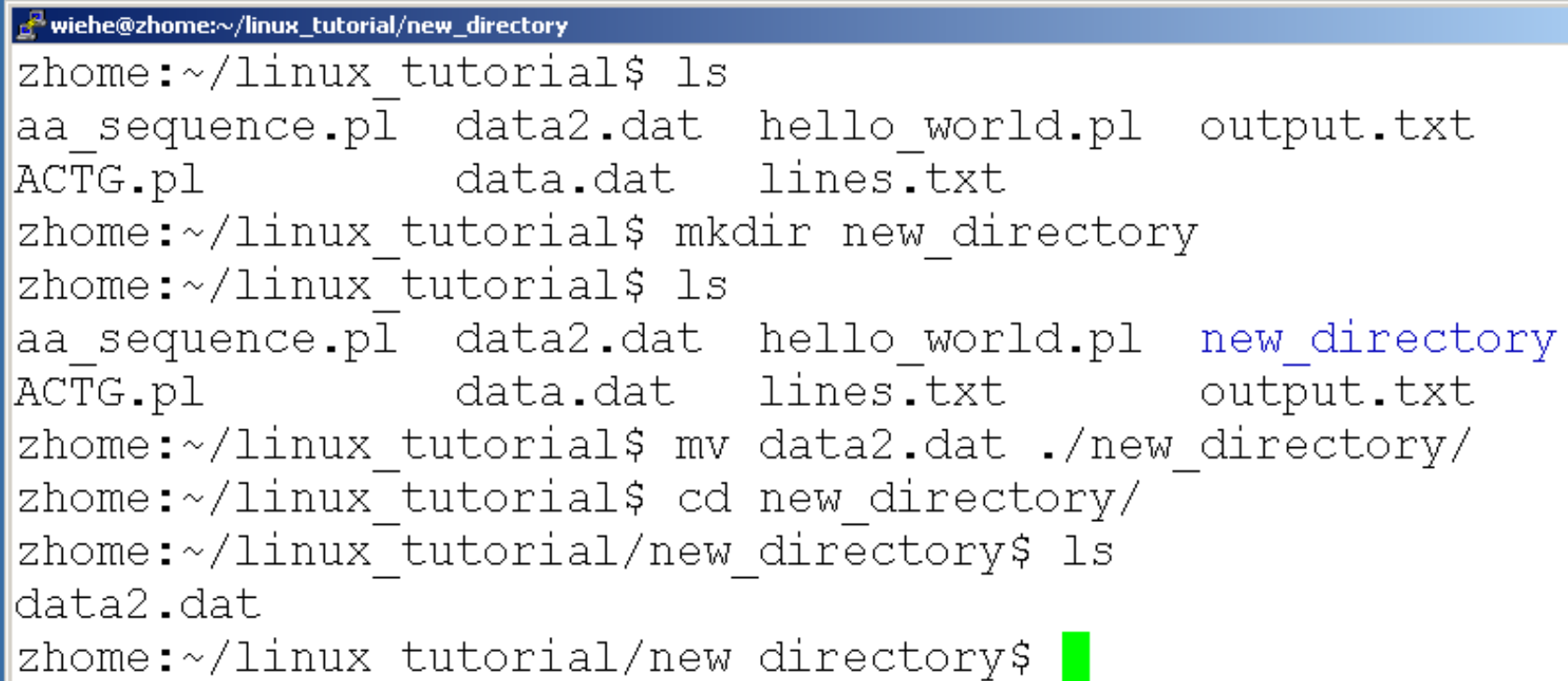
- To copy a file use “cp”



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat      lines.txt
ACTG.pl        hello_world.pl output.txt
zhome:~/linux_tutorial$ cp data.dat data2.dat
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data2.dat  hello_world.pl  output.txt
ACTG.pl        data.dat   lines.txt
zhome:~/linux_tutorial$ █
```

# Command: mv

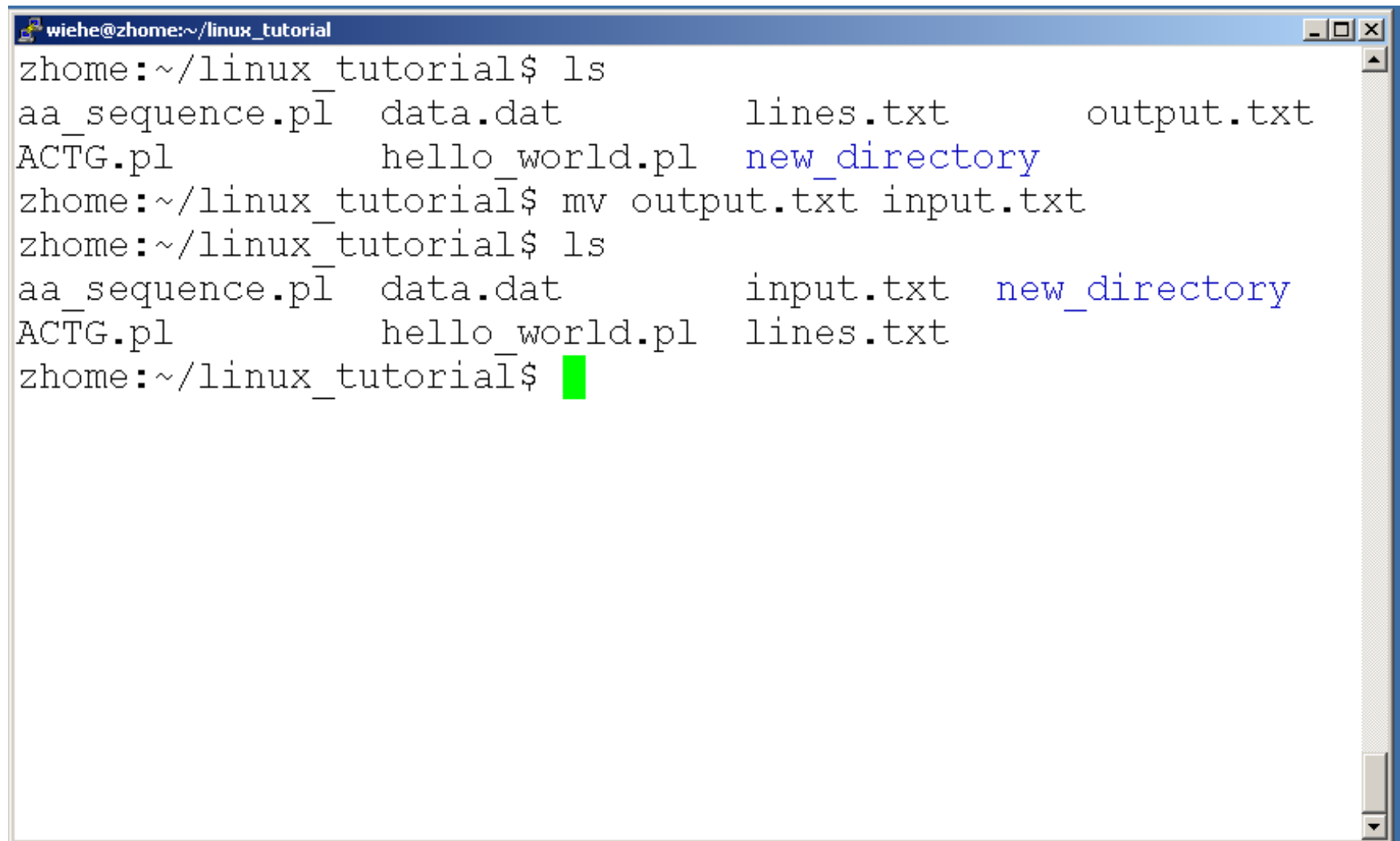
- To move a file to a different location use “mv”



```
wiehe@zhome:~/linux_tutorial/new_directory
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data2.dat  hello_world.pl  output.txt
ACTG.pl        data.dat   lines.txt
zhome:~/linux_tutorial$ mkdir new_directory
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data2.dat  hello_world.pl  new_directory
ACTG.pl        data.dat   lines.txt       output.txt
zhome:~/linux_tutorial$ mv data2.dat ./new_directory/
zhome:~/linux_tutorial$ cd new_directory/
zhome:~/linux_tutorial/new_directory$ ls
data2.dat
zhome:~/linux_tutorial/new_directory$ █
```

# Command: mv

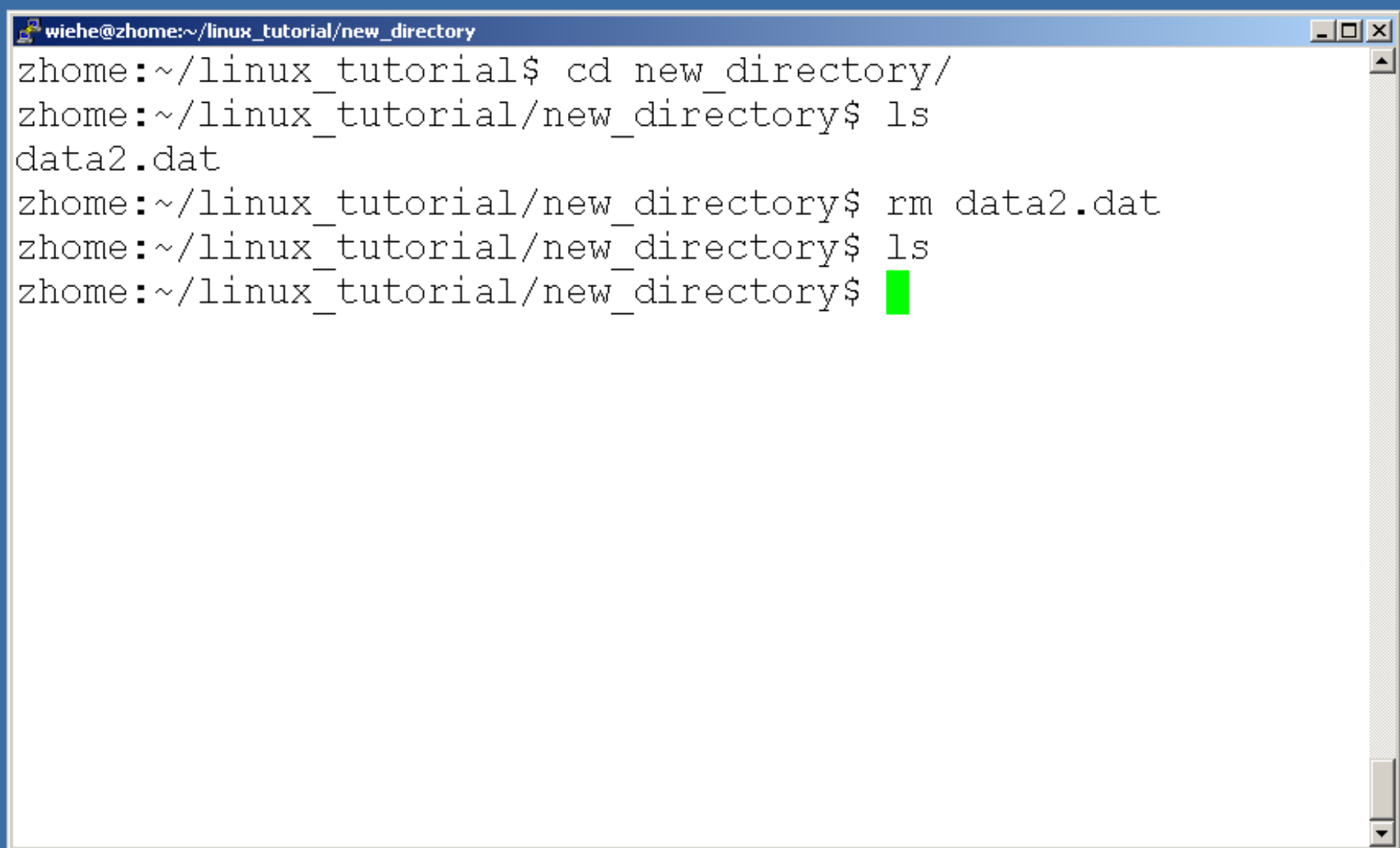
- mv can also be used to rename a file



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          lines.txt         output.txt
ACTG.pl        hello_world.pl   new_directory
zhome:~/linux_tutorial$ mv output.txt input.txt
zhome:~/linux_tutorial$ ls
aa_sequence.pl  data.dat          input.txt         new_directory
ACTG.pl        hello_world.pl   lines.txt
zhome:~/linux_tutorial$ █
```

# Command: rm

- To remove a file use “rm”



```
wiehe@zhome:~/linux_tutorial/new_directory
zhome:~/linux_tutorial$ cd new_directory/
zhome:~/linux_tutorial/new_directory$ ls
data2.dat
zhome:~/linux_tutorial/new_directory$ rm data2.dat
zhome:~/linux_tutorial/new_directory$ ls
zhome:~/linux_tutorial/new_directory$ █
```



# Command: rm

- To remove a file “recursively”: `rm -r`
- Used to remove all files and directories
- Be very careful, deletions are permanent in Unix/Linux

# File permissions

- Each file in Unix/Linux has an associated permission level
- This allows the user to prevent others from reading/writing/executing their files or directories
- Use “`ls -l filename`” to find the permission level of that file

# Permission levels

- “r” means “read only” permission
- “w” means “write” permission
- “x” means “execute” permission
  - In case of directory, “x” grants permission to list directory contents

# File Permissions

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls -l
total 28
-rw-rw-r--  1 wiehe wiehe  169 Aug 30 12:20 aa_sequence.pl
-rw-rw-r--  1 wiehe wiehe   92 Aug 30 11:54 ACTG.pl
-rw-rw-r--  1 wiehe wiehe   21 Aug 30 12:23 data.dat
-rw-rw-r--  1 wiehe wiehe   42 Aug 30 12:22 hello_world.pl
-rw-rw-r--  1 wiehe wiehe   24 Aug 30 12:23 input.txt
-rw-rw-r--  1 wiehe wiehe   50 Aug 30 13:13 lines.txt
drwxrwxr-x  2 wiehe wiehe 4096 Aug 30 13:19 new_directory
zhome:~/linux_tutorial$ █
```

**User (you)**

# File Permissions

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls -l
total 28
-rw-rw-r--  1 wiehe wiehe  169 Aug 30 12:20 aa_sequence.pl
-rw-rw-r--  1 wiehe wiehe   92 Aug 30 11:54 ACTG.pl
-rw-rw-r--  1 wiehe wiehe   21 Aug 30 12:23 data.dat
-rw-rw-r--  1 wiehe wiehe   42 Aug 30 12:22 hello_world.pl
-rw-rw-r--  1 wiehe wiehe   24 Aug 30 12:23 input.txt
-rw-rw-r--  1 wiehe wiehe   50 Aug 30 13:13 lines.txt
drwxrwxr-x  2 wiehe wiehe 4096 Aug 30 13:19 new_directory
zhome:~/linux_tutorial$
```

**Group**

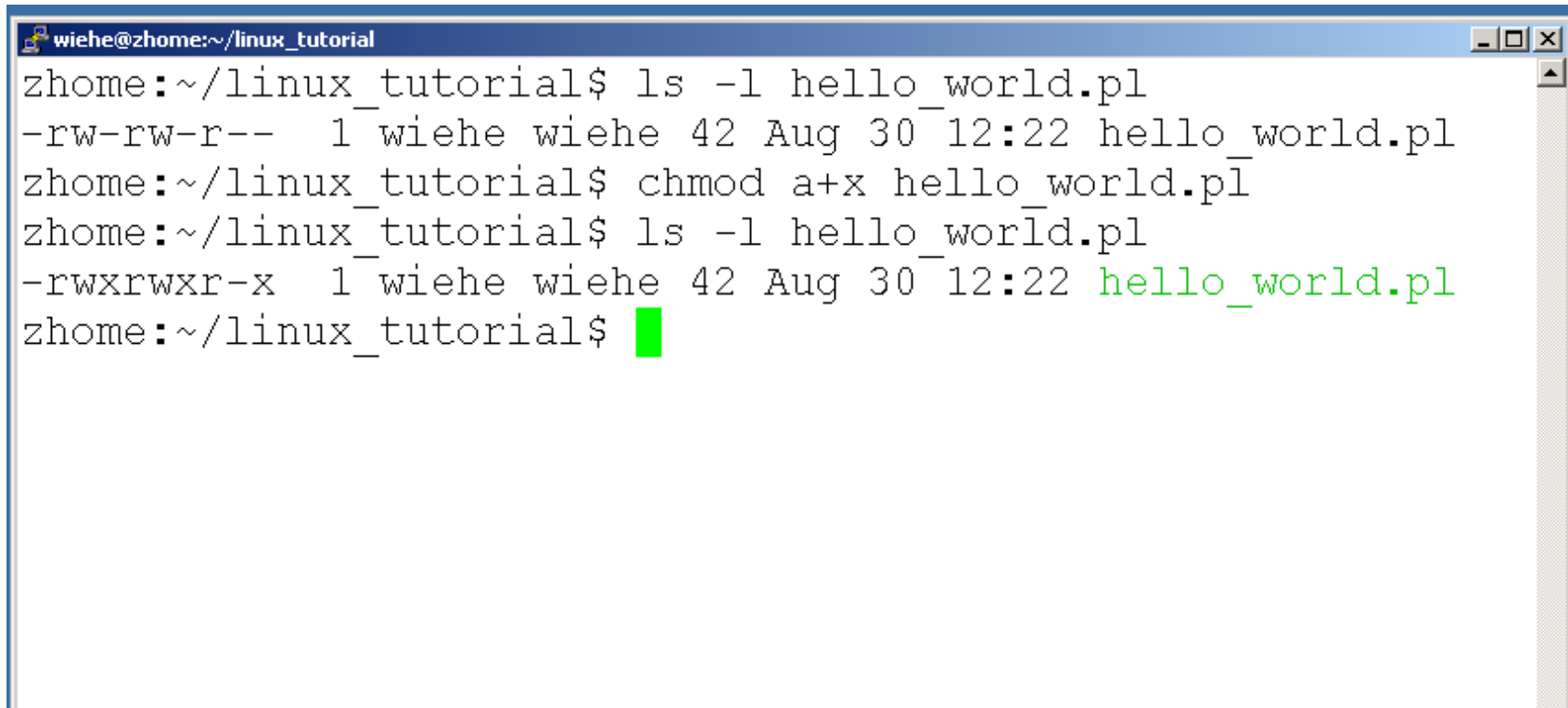
# File Permissions

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls -l
total 28
-rw-rw-r-- 1 wiehe wiehe 169 Aug 30 12:20 aa_sequence.pl
-rw-rw-r-- 1 wiehe wiehe 92 Aug 30 11:54 ACTG.pl
-rw-rw-r-- 1 wiehe wiehe 21 Aug 30 12:23 data.dat
-rw-rw-r-- 1 wiehe wiehe 42 Aug 30 12:22 hello_world.pl
-rw-rw-r-- 1 wiehe wiehe 24 Aug 30 12:23 input.txt
-rw-rw-r-- 1 wiehe wiehe 50 Aug 30 13:13 lines.txt
drwxrwxr-x 2 wiehe wiehe 4096 Aug 30 13:19 new_directory
zhome:~/linux_tutorial$
```

**“The World”**

# Command: chmod

- If you own the file, you can change it's permissions with "chmod"
  - Syntax: chmod [**u**ser/**g**roup/**o**thers/**a**ll]+[permission] [file(s)]
  - Below we grant execute permission to all:

A terminal window titled 'wiehe@zhome:~/linux\_tutorial' showing the execution of the 'chmod' command. The window has a blue title bar and standard window controls. The terminal text is as follows:

```
wiehe@zhome:~/linux_tutorial$ ls -l hello_world.pl
-rw-rw-r-- 1 wiehe wiehe 42 Aug 30 12:22 hello_world.pl
wiehe@zhome:~/linux_tutorial$ chmod a+x hello_world.pl
wiehe@zhome:~/linux_tutorial$ ls -l hello_world.pl
-rwxrwxr-x 1 wiehe wiehe 42 Aug 30 12:22 hello_world.pl
wiehe@zhome:~/linux_tutorial$
```

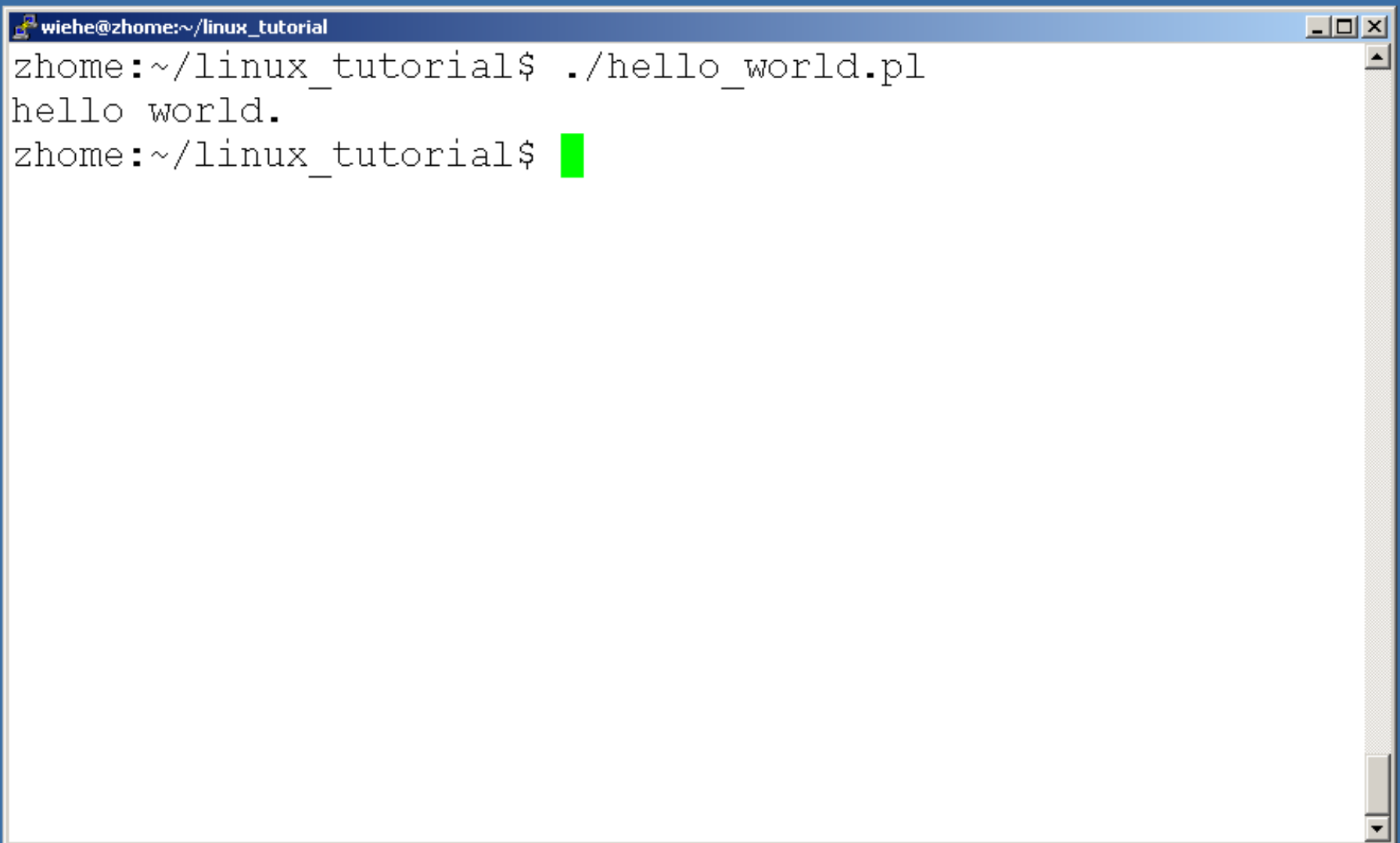
# Running a program (a.k.a. a job)

- Make sure the program has executable permissions
- Use “./” to run the program



# Running a program: an example

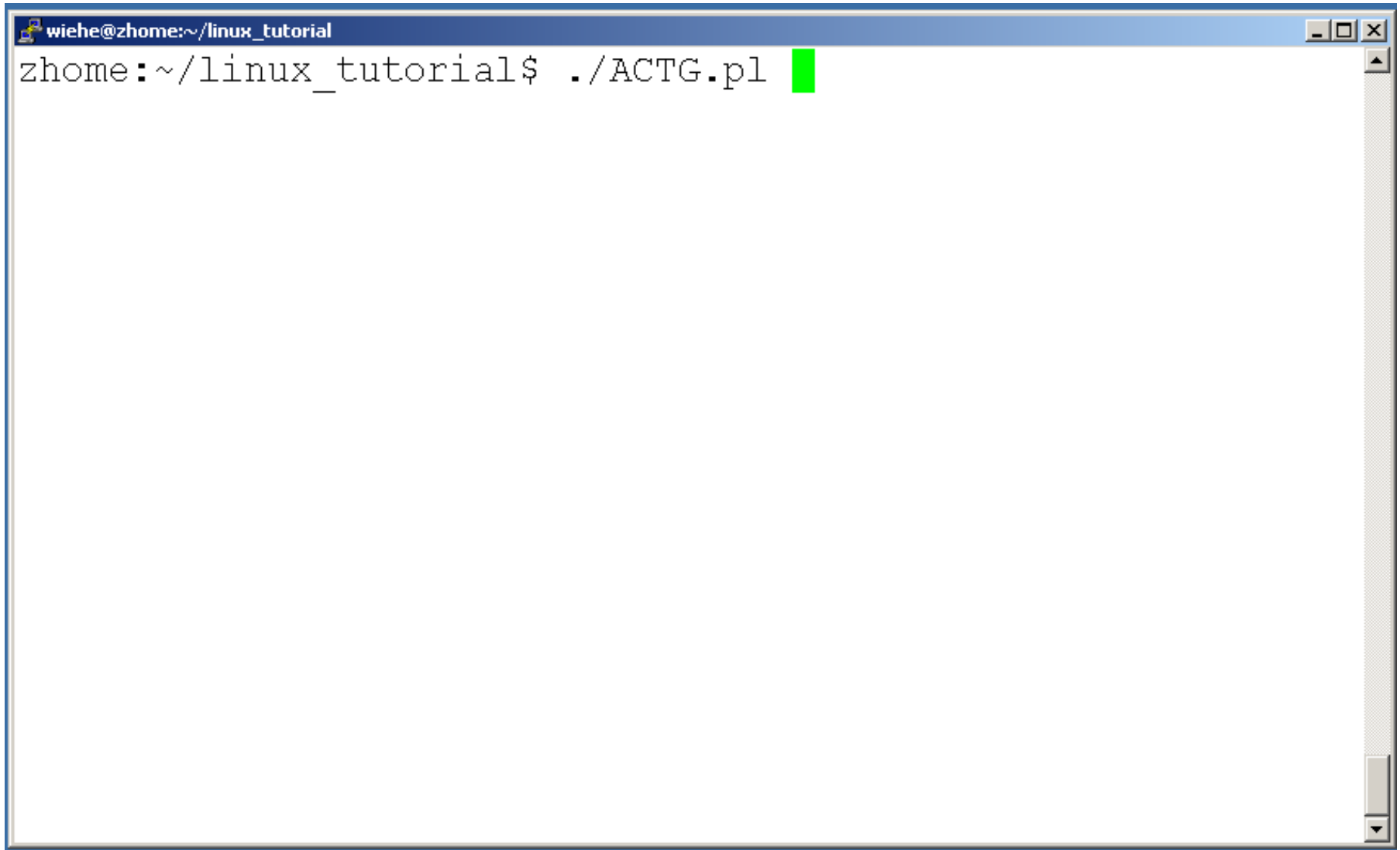
- Running the sample perl script “hello\_world.pl”

A terminal window with a blue title bar containing the text "wiehe@zhome:~/linux\_tutorial". The terminal content shows a shell prompt "zhome:~/linux\_tutorial\$" followed by the command "./hello\_world.pl". The output is "hello world." on the next line. A second shell prompt "zhome:~/linux\_tutorial\$" is shown with a green cursor block. The window has standard Linux window controls (minimize, maximize, close) in the top right corner and a vertical scrollbar on the right side.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ./hello_world.pl
hello world.
zhome:~/linux_tutorial$ █
```

# Ending a program

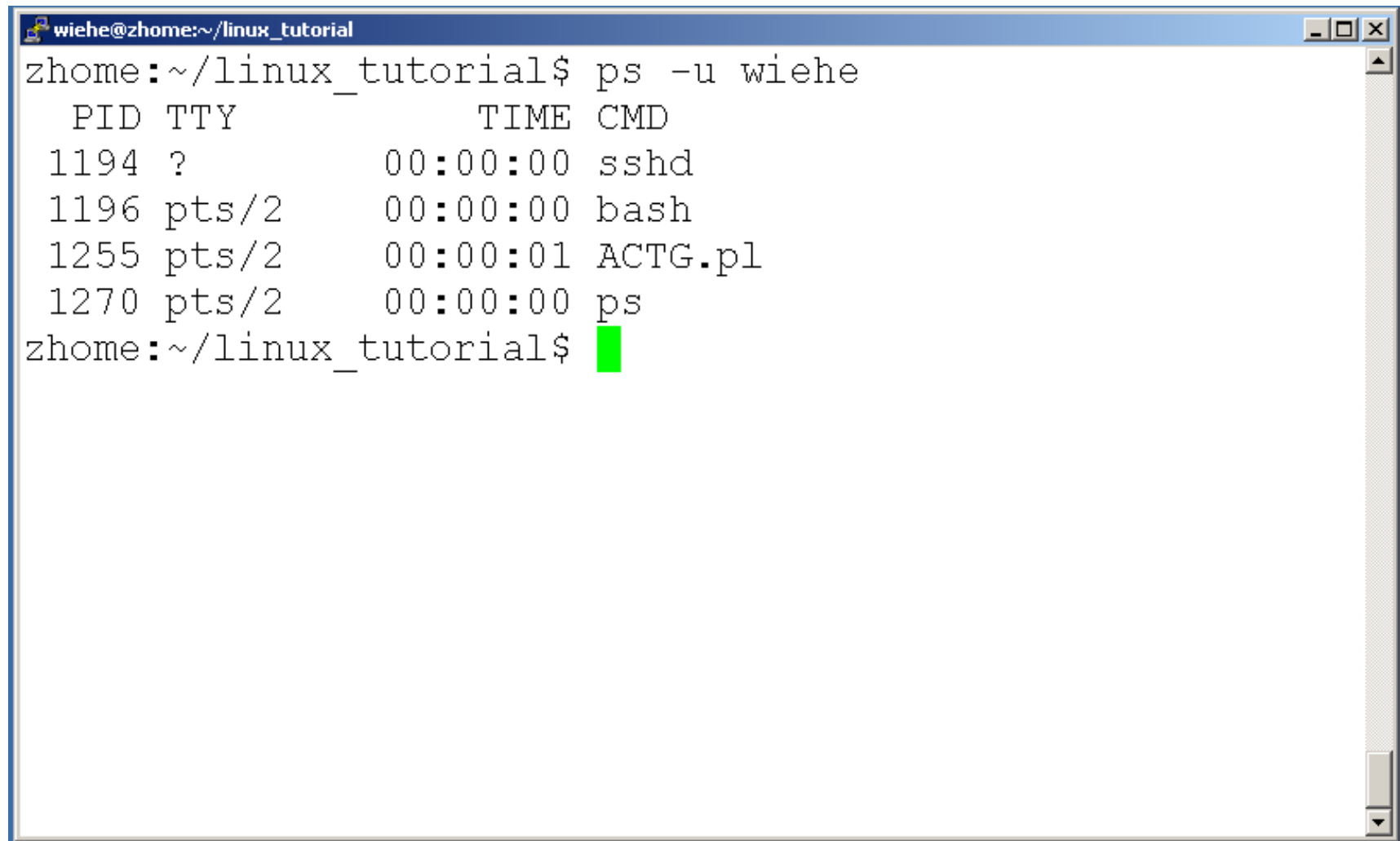
- To end a program use “ctrl-c”. To try it:

A screenshot of a Linux terminal window. The window title bar shows the user 'wiehe' at 'zhome' in the directory '~/linux\_tutorial'. The terminal content shows the command './ACTG.pl' being entered at the prompt 'zhome:~/linux\_tutorial\$'. A green cursor is positioned at the end of the command line.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ./ACTG.pl
```

# Command: ps

- To view the processes that you're running:



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ps -u wiehe
  PID TTY          TIME CMD
 1194 ?            00:00:00 sshd
 1196 pts/2        00:00:00 bash
 1255 pts/2        00:00:01 ACTG.pl
 1270 pts/2        00:00:00 ps
zhome:~/linux_tutorial$
```

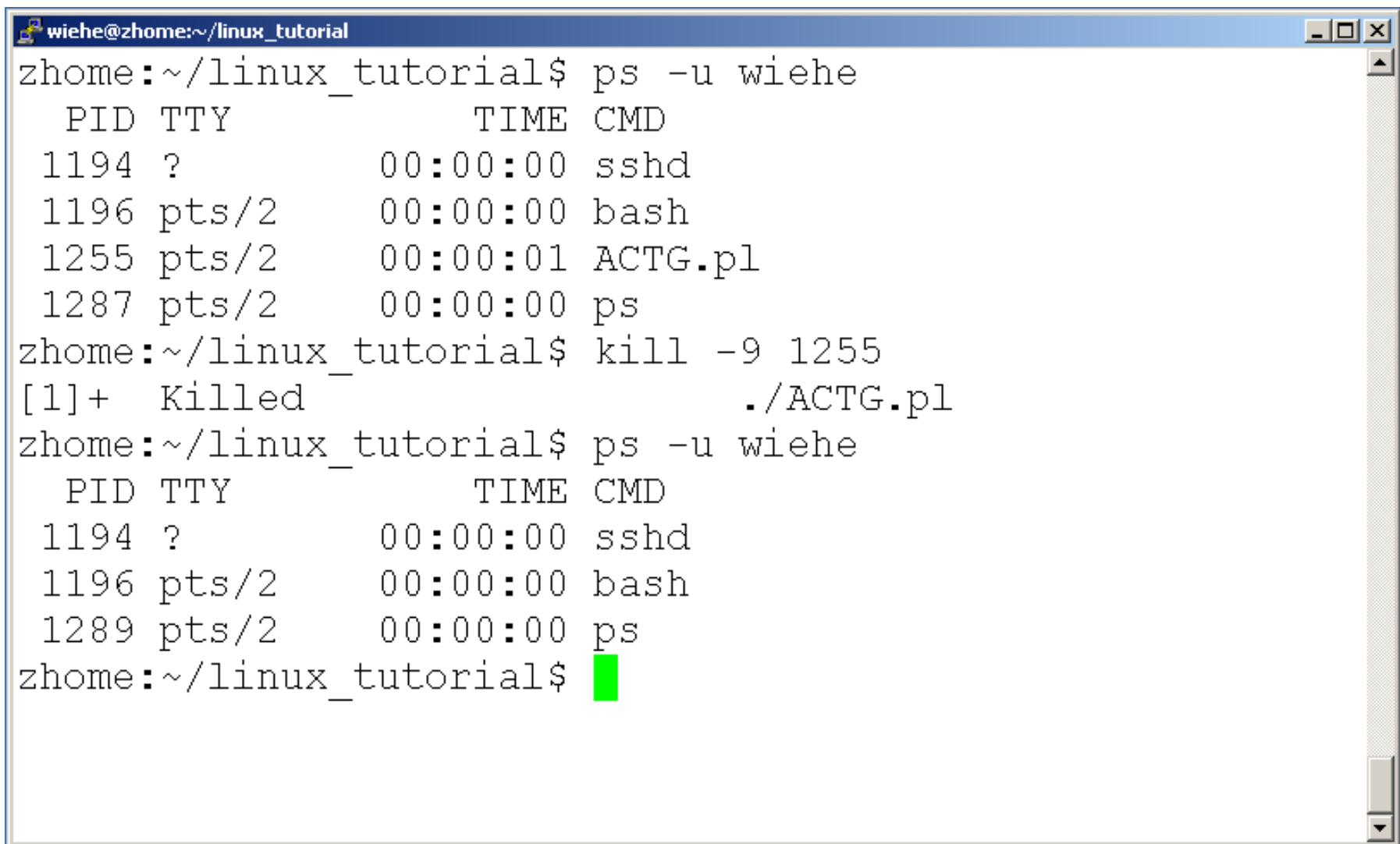
# Command: top

- To view the CPU usage of all processes:

```
wiehe@zhome:~/linux_tutorial
top - 13:46:33 up 50 days, 4:26, 2 users, load avera
Tasks:      total,      running,      sleeping,      stoppe
Cpu(s) :    us,      sy,      ni,      id,      w
Mem:        total,      used,      free,
Swap:       total,      used,      free,
█
  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM
 3403 root        15   0    0     0    0    0  S   0.7   0.0
   1  root        16   0 1604   324  292  S   0.0   0.0
   2  root        RT   0    0     0    0    0  S   0.0   0.0
   3  root        34  19    0     0    0    0  S   0.0   0.0
   4  root        RT   0    0     0    0    0  S   0.0   0.0
   5  root        34  19    0     0    0    0  S   0.0   0.0
   6  root        RT   0    0     0    0    0  S   0.0   0.0
   7  root        34  19    0     0    0    0  S   0.0   0.0
   8  root        RT   0    0     0    0    0  S   0.0   0.0
   9  root        34  19    0     0    0    0  S   0.0   0.0
```

# Command: kill

- To terminate a process use “kill”

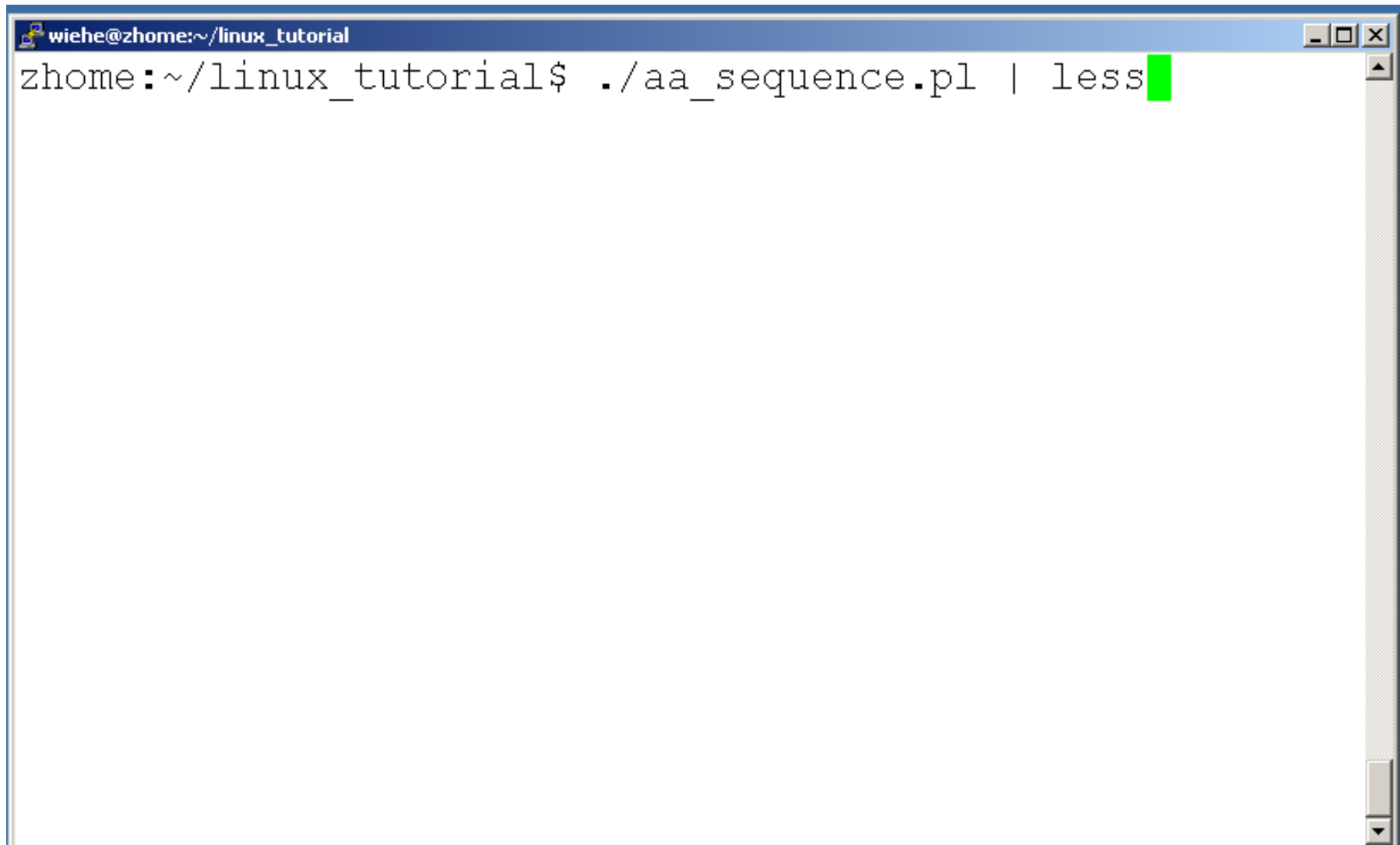


```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ps -u wiehe
  PID TTY          TIME CMD
 1194 ?            00:00:00 sshd
 1196 pts/2        00:00:00 bash
 1255 pts/2        00:00:01 ACTG.pl
 1287 pts/2        00:00:00 ps
zhome:~/linux_tutorial$ kill -9 1255
[1]+  Killed                  ./ACTG.pl
zhome:~/linux_tutorial$ ps -u wiehe
  PID TTY          TIME CMD
 1194 ?            00:00:00 sshd
 1196 pts/2        00:00:00 bash
 1289 pts/2        00:00:00 ps
zhome:~/linux_tutorial$ █
```

# Input/Output Redirection (“piping”)

- Programs can output to other programs
- Called “piping”
- “program\_a | program\_b”
  - program\_a’s output becomes program\_b’s input
- “program\_a > file.txt”
  - program\_a’s output is written to a file called “file.txt”
- “program\_a < input.txt”
  - program\_a gets its input from a file called “input.txt”

# A few examples of piping

A terminal window with a blue title bar containing the text 'wiehe@zhome:~/linux\_tutorial'. The main area of the terminal is white and contains the command 'zhome:~/linux\_tutorial\$ ./aa\_sequence.pl | less' followed by a green cursor. The window has standard Linux window controls (minimize, maximize, close) in the top right corner and a vertical scrollbar on the right side.

```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ./aa_sequence.pl | less
```

# A few examples of piping

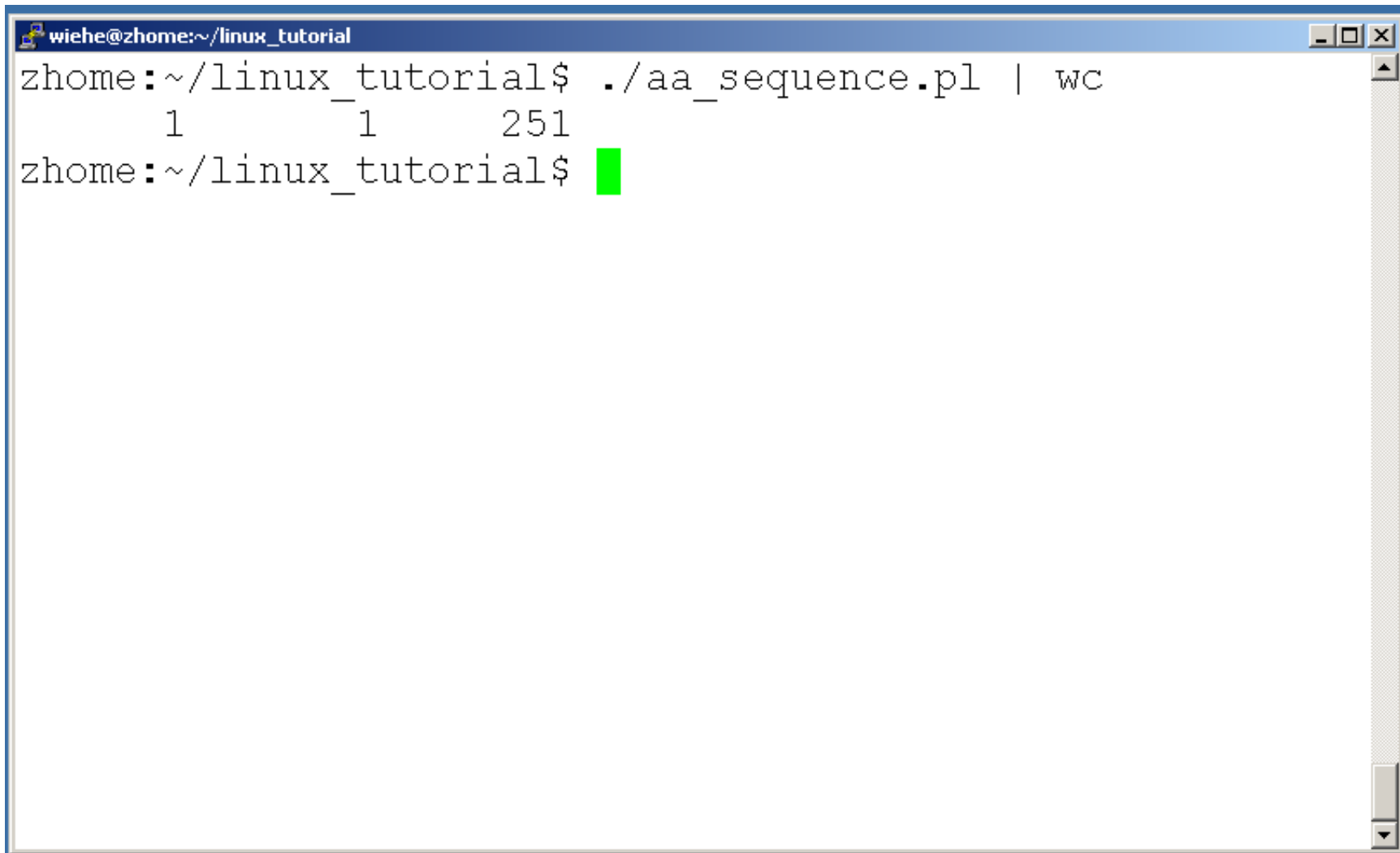
```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  hello_world.pl  new_directory
ACTG.pl        input.txt
data.dat       lines.txt
zhome:~/linux_tutorial$ ./aa_sequence.pl > sequence.txt
zhome:~/linux_tutorial$ ls
aa_sequence.pl  hello_world.pl  new_directory
ACTG.pl        input.txt       sequence.txt
data.dat       lines.txt
zhome:~/linux_tutorial$ less sequence.txt
```



# Command: `wc`

- To count the characters, words, and lines in a file use “`wc`”
- The first column in the output is lines, the second is words, and the last is characters

# A few examples of piping

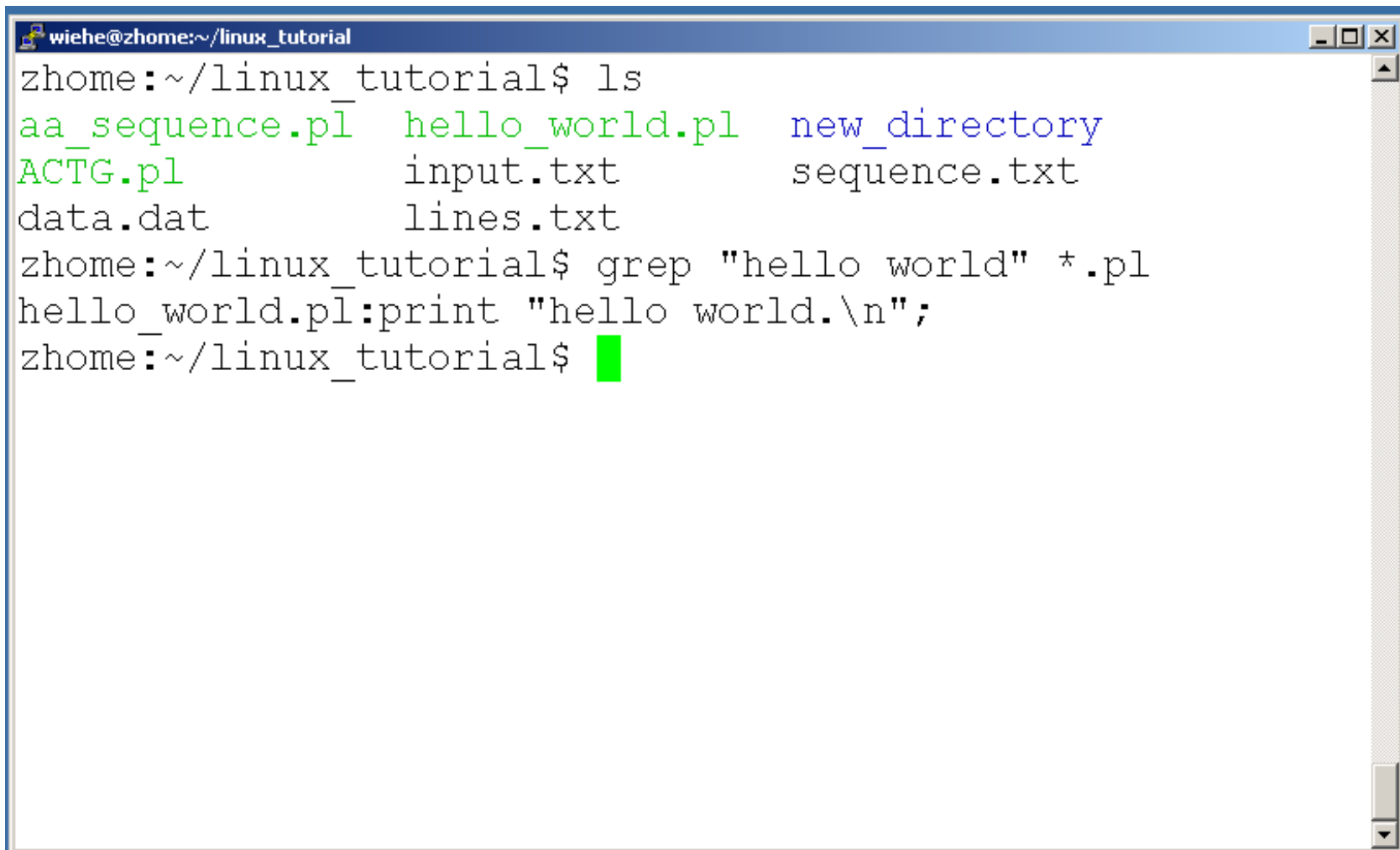


```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ./aa_sequence.pl | wc
      1      1     251
zhome:~/linux_tutorial$
```

The image shows a terminal window with a blue title bar containing the text "wiehe@zhome:~/linux\_tutorial". The terminal content shows a command being executed: `./aa_sequence.pl | wc`. The output of this command is displayed on the next line as `1 1 251`. The prompt `zhome:~/linux_tutorial$` is shown again on the following line, followed by a green cursor block.

# Command: grep

- To search files in a directory for a specific string use “grep”



```
wiehe@zhome:~/linux_tutorial
zhome:~/linux_tutorial$ ls
aa_sequence.pl  hello_world.pl  new_directory
ACTG.pl        input.txt       sequence.txt
data.dat       lines.txt
zhome:~/linux_tutorial$ grep "hello world" *.pl
hello_world.pl:print "hello world.\n";
zhome:~/linux_tutorial$
```

# Command: diff

- To compare to files for differences use “diff”
  - Try: `diff /dev/null hello.txt`
  - `/dev/null` is a special address -- it is always empty, and anything moved there is deleted

# ssh, scp

- ssh is used to securely log in to remote systems, successor to telnet
- ssh [username]@[hostname]
- Try:
  - **ssh yourusername@localhost**
  - Type “exit” to log out of session
- Scp is used to copy files to/from remote systems, syntax is similar to cp:
  - scp [local path] [username]@[hostname]:[remote file path]
- Try:
  - **scp hello.txt yourusername@localhost:scp-test.txt**

# Unix Web Resources

- <http://www.ee.surrey.ac.uk/Teaching/Unix/>
- <http://www.ugu.com/sui/ugu/show?help.beginners>
- <http://en.wikipedia.org/wiki/Unix>

# Useful linux commands

command	description
pwd	Print the name of the current directory
ls	show the files in a directory
man	Open the manual page for a certain command
cd	Change directory
file	Examine a file
more	Print the contents of a text file (try also 'less')
perl	Run a perl script
ps	List the running processes
top	List the active processes, updating
kill	Terminate a process try also 'nkill' and

# and more...

command	description
exit	Close the current shell (also ctrl+d)
ssh	Open a connection to a remote computer
cp	Copy a file to a new location
mv	Move a file to a new location (or rename a file)
rm	Remove/delete a file
mkdir	Create a new directory
rmdir	Remove a directory
echo	Print a text
find	Find a file
grep	Find a text in one or more files



# and even more...

command	description
screen	start a virtual terminal, if you want to switch off PC
scp	secure network copy
cat	print the contents of a file to the terminal
watch	monitor the output of a shell command
tail	print the last lines of a text file, continuously
head	print the first lines of a text file
tar	create or read a tarball archive
gzip	zip a file, use gunzip to unzip
date	print current date/time
ln	create a (soft) link to a file, like a shortcut

...

command	description
su	switch to another user, default is 'root'
sudo	act as another user, password required
zypper	install new software, other flavours: apt-get, yum
reboot	reboot Linux
shutdown	shutdown the Linux environment
printenv	print the environment variables
setenv	set the environment variables, depends on shell
passwd	change your password