

Homework5

April 6, 2020

0.1 Homework#5

Permission

0 -- 000 - Do nothing

1 -- 001 - Execute

2 -- 010 - Write

4 -- 100 - Read

```
[5]: cd ..
```

```
[6]: cd homework5
```

```
[7]: touch 1.txt
```

```
[9]: ls -l
```

```
total 0
```

```
-rw-r--r-- 1 jupyter-zhannaspac jupyter-zhannaspac 0 Apr  6 18:30 1.txt
```

```
[10]: chmod 501 1.txt  # for user: rx, for group: nothing, for others: only execute
```

```
[11]: ls -l 1.txt
```

```
-r-x-----x 1 jupyter-zhannaspac jupyter-zhannaspac 0 Apr  6 18:30 1.txt
```

```
[13]: chmod 777 1.txt #give all rights to all users
```

```
[14]: ls -l
```

```
total 0
```

```
-rwxrwxrwx 1 jupyter-zhannaspac jupyter-zhannaspac 0 Apr  6 18:30 1.txt
```

```
[15]: chmod 621 1.txt # user: wr,group:w, others : x
```

```
[16]: ls -l
```

```
total 0
```

```
-rw--w---x 1 jupyter-zhannaspac jupyter-zhannaspac 0 Apr  6 18:30 1.txt
```

```
[17]: mkdir zhanna
```

```
[18]: ls -ld zhanna
```

```
drwxr-xr-x 2 jupyter-zhannaspac jupyter-zhannaspac 4096 Apr  6 18:41 zhanna
```

```
[21]: chmod 700 zhanna #user: all rights, g&O - nothing
```

```
[23]: ls -ld zhanna
```

```
drwx----- 2 jupyter-zhannaspac jupyter-zhannaspac 4096 Apr  6 18:41 zhanna
```

```
[24]: chmod 745 zhanna # user-all, g -r, o -rx
```

```
[25]: ls -ld zhanna
```

```
drwxr--r-x 2 jupyter-zhannaspac jupyter-zhannaspac 4096 Apr  6 18:41 zhanna
```

```
[1]: whoami
```

```
jupyter-zhannaspac
```

```
[3]: pwd
```

```
/home/jupyter-zhannaspac/homework5
```

0.2 Filter (head)

```
[6]: echo '1. Xi Jinping China 66  
2. Vladimir Putin Russia 67  
3. Donald Trump United States 73  
4. Angela Merkel Germany 65  
5. Jeff Bezos Amazon.com 56  
6. Pope Francis Roman Catholic Church 83  
7. Bill Gates Bill & Melinda Gates Foundation 64  
8. Mohammed bin Salman Al Saud Saudi Arabia 34  
9. Narendra Modi India 69  
10. Larry Page Alphabet 47  
11. Jerome H. Powell United States 67'> forbes.txt
```

```
[7]: ls
```

```
1.txt forbes forbes.txt Homework5.ipynb zhanna
```

```
[8]: head forbes.txt #show fist 10 lines
```

1. Xi Jinping China 66
2. Vladimir Putin Russia 67
3. Donald Trump United States 73
4. Angela Merkel Germany 65
5. Jeff Bezos Amazon.com 56
6. Pope Francis Roman Catholic Church 83
7. Bill Gates Bill & Melinda Gates Foundation 64
8. Mohammed bin Salman Al Saud Saudi Arabia 34
9. Narendra Modi India 69
10. Larry Page Alphabet 47

```
[9]: head -5 forbes.txt #show first 5 lines
```

1. Xi Jinping China 66
2. Vladimir Putin Russia 67
3. Donald Trump United States 73
4. Angela Merkel Germany 65
5. Jeff Bezos Amazon.com 56

```
[10]: head -3 forbes.txt
```

1. Xi Jinping China 66
2. Vladimir Putin Russia 67
3. Donald Trump United States 73

```
[11]: head -20 forbes.txt
```

1. Xi Jinping China 66
2. Vladimir Putin Russia 67
3. Donald Trump United States 73
4. Angela Merkel Germany 65
5. Jeff Bezos Amazon.com 56
6. Pope Francis Roman Catholic Church 83
7. Bill Gates Bill & Melinda Gates Foundation 64
8. Mohammed bin Salman Al Saud Saudi Arabia 34
9. Narendra Modi India 69
10. Larry Page Alphabet 47
11. Jerome H. Powell United States 67

```
[12]: tail forbes.txt #show last 10 lines
```

2. Vladimir Putin Russia 67
3. Donald Trump United States 73
4. Angela Merkel Germany 65
5. Jeff Bezos Amazon.com 56
6. Pope Francis Roman Catholic Church 83
7. Bill Gates Bill & Melinda Gates Foundation 64
8. Mohammed bin Salman Al Saud Saudi Arabia 34

```
9. Narendra Modi India 69
10. Larry Page Alphabet 47
11. Jerome H. Powell United States 67
```

```
[13]: tail -3 forbes. #show last 3 lines
```

```
9. Narendra Modi India 69
10. Larry Page Alphabet 47
11. Jerome H. Powell United States 67
```

```
[14]: cat forbes.txt #show all content of the file
```

```
1. Xi Jinping China 66
2. Vladimir Putin Russia 67
3. Donald Trump United States 73
4. Angela Merkel Germany 65
5. Jeff Bezos Amazon.com 56
6. Pope Francis Roman Catholic Church 83
7. Bill Gates Bill & Melinda Gates Foundation 64
8. Mohammed bin Salman Al Saud Saudi Arabia 34
9. Narendra Modi India 69
10. Larry Page Alphabet 47
11. Jerome H. Powell United States 67
```

```
[17]: man sort
```

```
SORT(1)                                User Commands                                SORT(1)

NAME
    sort - sort lines of text files

SYNOPSIS
    sort [OPTION]... [FILE]...
    sort [OPTION]... --files0-from=F

DESCRIPTION
    Write sorted concatenation of all FILE(s) to standard output.

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

    Mandatory arguments to long options are mandatory for short options
    too. Ordering options:

    -b, --ignore-leading-blanks
        ignore leading blanks

    -d, --dictionary-order
        consider only blanks and alphanumeric characters
```

-f, --ignore-case
 fold lower case to upper case characters

-g, --general-numeric-sort
 compare according to general numerical value

-i, --ignore-nonprinting
 consider only printable characters

-M, --month-sort
 compare (unknown) < 'JAN' < ... < 'DEC'

-h, --human-numeric-sort
 compare human readable numbers (e.g., 2K 1G)

-n, --numeric-sort
 compare according to string numerical value

-R, --random-sort
 shuffle, but group identical keys. See shuf(1)

--random-source=FILE
 get random bytes from FILE

-r, --reverse
 reverse the result of comparisons

--sort=WORD
 sort according to WORD: general-numeric -g, human-numeric -h,
 month -M, numeric -n, random -R, version -V

-V, --version-sort
 natural sort of (version) numbers within text

Other options:

--batch-size=NMERGE
 merge at most NMERGE inputs at once; for more use temp files

-c, --check, --check=diagnose-first
 check for sorted input; do not sort

-C, --check=quiet, --check=silent
 like -c, but do not report first bad line

--compress-program=PROG
 compress temporaries with PROG; decompress them with PROG -d

`--debug`
 annotate the part of the line used to sort, and warn about questionable usage to `stderr`

`--files0-from=F`
 read input from the files specified by NUL-terminated names in file `F`; If `F` is `-` then read names from standard input

`-k, --key=KEYDEF`
 sort via a key; `KEYDEF` gives location and type

`-m, --merge`
 merge already sorted files; do not sort

`-o, --output=FILE`
 write result to `FILE` instead of standard output

`-s, --stable`
 stabilize sort by disabling last-resort comparison

`-S, --buffer-size=SIZE`
 use `SIZE` for main memory buffer

`-t, --field-separator=SEP`
 use `SEP` instead of non-blank to blank transition

`-T, --temporary-directory=DIR`
 use `DIR` for temporaries, not `$TMPDIR` or `/tmp`; multiple options specify multiple directories

`--parallel=N`
 change the number of sorts run concurrently to `N`

`-u, --unique`
 with `-c`, check for strict ordering; without `-c`, output only the first of an equal run

`-z, --zero-terminated`
 line delimiter is NUL, not newline

`--help` display this help and exit

`--version`
 output version information and exit

`KEYDEF` is `F[.C][OPTS][,F[.C][OPTS]]` for start and stop position, where `F` is a field number and `C` a character position in the field; both are

origin 1, and the stop position defaults to the line's end. If neither -t nor -b is in effect, characters in a field are counted from the beginning of the preceding whitespace. OPTS is one or more single-letter ordering options [bdfgiMhnRrV], which override global ordering options for that key. If no key is given, use the entire line as the key. Use --debug to diagnose incorrect key usage.

SIZE may be followed by the following multiplicative suffixes: % 1% of memory, b 1, K 1024 (default), and so on for M, G, T, P, E, Z, Y.

*** WARNING *** The locale specified by the environment affects sort order. Set LC_ALL=C to get the traditional sort order that uses native byte values.

AUTHOR

Written by Mike Haertel and Paul Eggert.

REPORTING BUGS

GNU coreutils online help: <<http://www.gnu.org/software/coreutils/>>
Report sort translation bugs to <<http://translationproject.org/team/>>

COPYRIGHT

Copyright © 2017 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <<http://gnu.org/licenses/gpl.html>>. This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

SEE ALSO

shuf(1), uniq(1)

Full documentation at: <<http://www.gnu.org/software/coreutils/sort>>
or available locally via: info '(coreutils) sort invocation'

GNU coreutils 8.28

January 2018

SORT(1)

```
[20]: echo 'anna
ben
conny
dan
eva
frank
igor
gemma
zak
tom
sid'> names.txt
```

```
[21]: cat names.txt
```

```
anna  
ben  
conny  
dan  
eva  
frank  
igor  
gemma  
zak  
tom  
sid
```

```
[22]: sort names.txt #sort in alphabet order
```

```
anna  
ben  
conny  
dan  
eva  
frank  
gemma  
igor  
sid  
tom  
zak
```

```
[2]: sort -r names.txt #sort in reverse alphabet order
```

```
zak  
tom  
sid  
igor  
gemma  
frank  
eva  
dan  
conny  
ben  
anna
```

```
[3]: nl names.txt #show number of lines
```

```
1 anna  
2 ben  
3 conny  
4 dan
```



```
5 eva
6 frank
7 igor
8 gemma
9 zak
10 tom
11 sid
```

```
[9]: nl -s'. ' -w 10 names.txt
```

```
1. anna
2. ben
3. conny
4. dan
5. eva
6. frank
7. igor
8. gemma
9. zak
10. tom
11. sid
```

```
[10]: nl -s'. ' -w 7 names.txt
```

```
1. anna
2. ben
3. conny
4. dan
5. eva
6. frank
7. igor
8. gemma
9. zak
10. tom
11. sid
```

```
[11]: wc names.txt #count number of lines(1) words(2) characters(3)
```

```
11 11 52 names.txt
```

```
[12]: wc -l names.txt #count number of lines
```

```
11 names.txt
```

```
[13]: wc -w names.txt #count number of words
```

```
11 names.txt
```

```
[14]: wc -m names.txt #count number of characters
```

```
52 names.txt
```

```
[18]: man cut
```

```
CUT(1) User Commands CUT(1)
```

NAME

cut - remove sections from each line of files

SYNOPSIS

cut OPTION... [FILE]...

DESCRIPTION

Print selected parts of lines from each FILE to standard output.

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

Mandatory arguments to long options are mandatory for short options too.

-b, --bytes=LIST

select only these bytes

-c, --characters=LIST

select only these characters

-d, --delimiter=DELIM

use DELIM instead of TAB for field delimiter

-f, --fields=LIST

select only these fields; also print any line that contains no delimiter character, unless the -s option is specified

-n (ignored)

--complement

complement the set of selected bytes, characters or fields

-s, --only-delimited

do not print lines not containing delimiters

--output-delimiter=STRING

use STRING as the output delimiter the default is to use the input delimiter

-z, --zero-terminated

line delimiter is NUL, not newline

--help display this help and exit

--version

output version information and exit

Use one, and only one of -b, -c or -f. Each LIST is made up of one range, or many ranges separated by commas. Selected input is written in the same order that it is read, and is written exactly once. Each range is one of:

N N'th byte, character or field, counted from 1

N- from N'th byte, character or field, to end of line

N-M from N'th to M'th (included) byte, character or field

-M from first to M'th (included) byte, character or field

AUTHOR

Written by David M. Ihnat, David MacKenzie, and Jim Meyering.

REPORTING BUGS

GNU coreutils online help: <<http://www.gnu.org/software/coreutils/>>

Report cut translation bugs to <<http://translationproject.org/team/>>

COPYRIGHT

Copyright © 2017 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <<http://gnu.org/licenses/gpl.html>>.

This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

SEE ALSO

Full documentation at: <<http://www.gnu.org/software/coreutils/cut>>
or available locally via: info '(coreutils) cut invocation'

GNU coreutils 8.28

January 2018

CUT(1)

[24]: `cut -f 1-3 -d ' ' forbes.txt # print only first 3 columns`

1. Xi Jinping
2. Vladimir Putin
3. Donald Trump
4. Angela Merkel
5. Jeff Bezos
6. Pope Francis
7. Bill Gates

8. Mohammed bin
9. Narendra Modi
10. Larry Page
11. Jerome H.

```
[27]: cut -f 1,4 -d ' ' forbes.txt #print only 1 and 4 columns
```

1. China
2. Russia
3. United
4. Germany
5. Amazon.com
6. Roman
7. Bill
8. Salman
9. India
10. Alphabet
11. Powell

```
[39]: echo ' anna  
ted  
anna  
anna  
maison  
nora'>dupl.txt
```

```
[40]: cat dupl.txt
```

```
anna  
ted  
anna  
anna  
maison  
nora
```

```
[41]: uniq dupl.txt
```

```
anna  
ted  
anna  
maison  
nora
```

```
[43]: cat names.txt
```

```
anna  
ben  
conny
```

```
dan  
eva  
frank  
igor  
gemma  
zak  
tom  
sid
```

```
[42]: tac names.txt
```

```
sid  
tom  
zak  
gemma  
igor  
frank  
eva  
dan  
conny  
ben  
anna
```

```
[ ]:
```