

11. [65/127] If I am in directory `/tmp` and `dir` is an empty sub-directory, what is true after this command line:
- ```
touch cow ; mkdir pig ; mv cow pig/dir
```
- the directory `pig` now contains a file named `cow`
  - the directory `dir` now contains a directory named `pig`
  - the directory `dir` is still empty
  - the directory `dir` now contains a file named `cow`
  - the command fails because `pig/dir` is not a directory
12. [65/124] If my current directory is `/etc`, which of these pathnames is equivalent to the file name `/etc/passwd`?
- `../../etc/passwd`
  - `/root/etc/passwd`
  - `../etc/passwd/`
  - `./etc/passwd`
  - `passwd/`
13. [66/123] What is the correct syntax to redirect both standard output and standard error into the same output file?
- `cmd 2>1 >out`
  - `cmd >out 2>&1`
  - `cmd 1>out 2>out`
  - `cmd 2>&1 >out`
  - `cmd 1>out 2>1`
14. [66/126] If file `a` contains 2 lines, and file `b` contains 3 lines, then how many lines are in file `c` after this command line:
- ```
cat a b >c ; cat a >>b ; sort c b >c a
```
- 12
 - 8
 - 0
 - 7
 - 5
15. [68/125] If my current directory is `/foo`, which command copies the password file into existing directory `/foo/dir` under the name `bar`?
- `cp ../../etc/passwd ../dir/bar`
 - `cp dir/../../etc/passwd dir/bar`
 - `cp ../../etc/passwd /dir/bar`
 - `cp ../dir/../../etc/passwd ../foo/dir/bar`
 - `cp ../foo/./dir/../../etc/passwd ./dir/./bar`
16. [68/124] Which command line lists all possible utilities available for compiling programs?
- `grep 'compile' /etc/`
 - `man -k compile`
 - `man compile`
 - `find compile`
 - `locate compile`

17. [69/126] If my current directory is `/bin`, which of these pathnames is equivalent to the file name `/bin/ls`?
- `../bin/ls/`
 - `ls/`
 - `./bin/ls`
 - `../../bin/ls`
 - `/root/bin/ls`
18. [70/127] What is the output of this command line in an empty directory:
- ```
date >.a ; touch .b ; echo .*
```
- `. .. .a .b`
  - `.b`
  - an error message from `echo` saying `.*` does not exist
  - `.*`
  - `.a .b`
19. [70/124] If I am in directory `/tmp` and `foo` is an empty sub-directory, what is true after this command line:
- ```
touch foo/bar ; mkdir bar ; mv foo/bar bar/me
```
- the `mkdir` fails because `bar` already exists
 - the directory `foo` is now empty
 - the directory `foo` now contains only a file named `me`
 - the command fails because the name `bar/me` does not exist
 - there is a second copy of the file `bar` in the file named `me`
20. [72/126] In an empty directory, what is the output on your screen of this command line: `echo hi >foo >bar ; cat foo`
- `hi`
 - `hi >foo >bar`
 - `hi >foo`
 - `cat: foo: No such file or directory`
 - no output
21. [73/122] If file `nine` contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9), what is the output on your screen of this command: `sort nine nine | tail -n 3 | head -n 1`
- 8 8
 - 2 2
 - 9
 - 8
 - 1

22. [73/126] If I am in directory `/tmp` and `dir` is an empty sub-directory, what is true after this command line:
- ```
touch bar ; mkdir dog ; mv bar dir/dog
```
- the command fails because `dir/dog` is not a directory
  - the directory `dir` now contains a file named `bar`
  - the directory `dir` is still empty
  - the directory `dog` now contains a file named `bar`
  - the directory `dir` now contains a file named `dog`
23. [74/126] Give the minimum number of directories in this valid pathname:
- ```
/dog/cat/pig/cow/moo
```
- 5
 - 4
 - 2
 - 3
 - 6
24. [75/126] Which pathname almost always leads to the same file named:
- ```
/bin/ls
```
- `./bin/./ls`
  - `/bin/ls/.`
  - `./bin/ls`
  - `././bin/./ls`
  - `/bin/./bin/ls`
25. [76/126] In a manual page **SYNOPSIS** section, ellipsis (three dots) (...) mean:
- no special meaning
  - a hidden directory
  - the parent directory
  - something that is repeated
  - something that is optional
26. [76/124] What is the result of this exact command line:
- ```
cat /dog cat
```
- all the files under directory `/dog` with the name `cat` will be displayed
 - the two text strings `/dog` and `cat` will be displayed
 - file `/dog` will be copied to `cat`
 - the names of the pathnames `/dog` and `cat` will be displayed
 - the contents of the files `/dog` and `cat` will be displayed
27. [77/126] If file `dog` contains 3 lines, and file `cat` contains 4 lines, then how many lines are output on your screen by this command line:
- ```
cat dog | tail cat
```
- 3 followed by 4
  - 4 followed by 3
  - 3
  - 5
  - 4

28. [77/126] Which of these command lines will make file `foo` contain all of the content of file `a` followed by all of the content of file `b`?
- `cp a >foo ; cp b >>foo`
  - `mv a b >foo`
  - `cp a b >foo`
  - `echo a b >foo`
  - `cat a >foo ; cat b >>foo`
29. [77/125] Which command line displays all the names in the current directory that are exactly three digits long (and no other names)?
- `echo [?][?][?]`
  - `echo [1-3][1-3][1-3]`
  - `echo ???`
  - `echo [3][3][3]`
  - `echo [0-9][0-9][0-9]`
30. [79/126] Given the pathname `/usr/bin/foo`, the *basename* of this pathname is:
- `usr`
  - `foo`
  - `bin`
  - `/usr/bin`
  - `/`
31. [80/123] If I am in directory `/tmp` and `foo` is an empty sub-directory, what is true after this command line:
- ```
touch foo/bar ; mkdir foo/me ; cp foo/bar ./foo/./me
```
- the command fails because the name `foo/bar` does not exist
 - there is a second copy of the file `bar` in file `/tmp/me`
 - there is a second copy of the file `bar` in directory `foo`
 - the directory `foo` now contains only a file named `me`
 - the directory `foo` is now empty
32. [81/127] If you type the command `wc`, which *CTRL* key will send an **EOF** and take you back to the command prompt?
- `^U`
 - `^R`
 - `^C`
 - `^D`
 - `^E`
33. [81/127] How many arguments and options are there to the command:
- ```
ls -l ls wc
```
- Three command line arguments, one of which contains one option.
  - One option and three command names.
  - Two command line arguments and one option.
  - Three command line arguments with no options.
  - Three pathname arguments and no options.

34. [82/126] In an empty directory, what happens after this command line:  
`mkdir a b c ; mv a b c`
- the directories **a** and **b** are appended to the directory **c**
  - the directories **a**, **b**, and **c** are moved to the current directory
  - an error message: **mv: target 'c' is not a directory**
  - the directories **a** and **b** are moved into the directory **c**
  - the directories **a**, **b**, and **c** are moved to the directory **c**
35. [82/127] If file **foo** contains 3 lines, and file **bar** contains 4 lines, then how many lines are output on your screen by this command line:  
`cp foo bar | cat`
- 3 followed by 4
  - 3
  - 4
  - no output on screen
  - 4 followed by 3
36. [82/126] What is in file **a** after this command line:  
`echo 1 2 >a 3 4`
- `echo 1 2`
  - `1 2`
  - `1 2 3 4`
  - nothing (empty file)
  - `3 4`
37. [82/127] In a directory that contains only the file **dog**, what happens after this command: `cp dog cat`
- the command fails because **cat** is not a directory
  - the command fails because the name **cat** does not exist
  - there is only the file named **cat** in the directory now
  - there is a copy of the file named **dog** in the file named **cat**
  - an empty file named **cat** is created
38. [84/123] What is the result of this exact command line:  
`echo /pig cow`
- all the files under directory **/pig** with the name **cow** will be displayed
  - the contents of the files **/pig** and **cow** will be displayed
  - file **/pig** will be copied to **cow**
  - the names of the pathnames **/pig** and **cow** will be displayed
  - the two text strings **/pig** and **cow** will be displayed

39. [85/126] The shell expands a leading tilde (~) in a pathname (e.g. `~/foo`) to be:
- the parent directory
  - the current directory
  - the ROOT directory
  - the directory **/root**
  - your HOME directory
40. [85/125] Which Unix command line deletes a directory and everything inside it?
- `rm -r dir`
  - `rmdir -r dir`
  - `rm -all dir`
  - `deltree -all dir`
  - `rmdir -all dir`
41. [88/125] Which command line displays all the non-hidden names in the current directory that contain the letter **z** (and no other names)?
- `echo *z`
  - `echo z*`
  - `echo *z*`
  - `echo [z]`
  - `echo ?z?`
42. [94/125] How many words are in the file **a** after this command line:  
`echo one two >a ; echo me >a ; echo you >>a`
- 4
  - 1
  - 3
  - 0
  - 2
43. [95/124] Which command line displays all the non-hidden names in the current directory that contain the case-insensitive word **hi** (and no other names)?
- `echo ?[HhIiHhIi]?`
  - `echo *[hiHI]*`
  - `echo *(H,h,I,i)*`
  - `echo *[Hh][Ii]*`
  - `echo ?[HhIi]?`
44. [95/121] **Did you read all the words of the test instructions on page one?**
- Tak** (Yes - Polish)
  - Igen** (Yes - Hungarian)
  - Taip** (Yes - Lithuanian)
  - Jes** (Yes - Esperanto)
  - Sim** (Yes - Portuguese)
45. [96/126] The option to **ls** that shows hidden names is:
- `-l`
  - `-a`
  - `-i`
  - `-1`
  - `-h`
46. [108/124] To "throw away" (hide) standard error output of a command, use:
- `cmd 1>&2`
  - `cmd 1>/dev/sda1`
  - `cmd 2>/dev/sda1`
  - `cmd 2>&1`
  - `cmd 2>/dev/null`