Cut

The cut command in UNIX is a command for cutting out the sections from each line of files and writing the result to standard output. It can be used to cut parts of a line by byte position, character and field. Basically the cut command slices a line and extracts the text. It is necessary to specify options with commands otherwise it gives errors. If more than one file name is provided then data from each file is not preceded by its file name. The file should contain column formatted data, ie tabular data
File should be something like:

eno|ename|esal|eaddr|dept|gender
100|sunny|1000|mumbai|admin|female
200|bunny|2000|chennai|sales|male
300|chinny|3000|delhi|accounting|female
400|vinny|4000|hyderabad|admin|male
500|pinny|5000|mumbai|sales|female

Syntax: cut option file

Options:

1) b → To extract the specific bytes, you need to follow -b option with the list of byte numbers separated by comma. Range of bytes can also be specified using the hyphen(-). Tabs and backspaces are treated like as a character of 1 byte. Example: cut -b 1,2,4-7,emp.dat
2) c → To cut by character use the -c option. Example: cut -c 1,2,4-7,emp.dat
3) -f → Most unix files doesn't have fixed-length lines. To extract the useful information you need to cut by fields rather than columns. List of the fields number specified must be separated by comma or |.
   Syntax: $cut -d "delimiter" -f (field number) file.txt. Example: cut -d '|' -f 3 emp.dat

Paste

We can use paste command to join two or more files horizontally by using some delimiter. Default delimiter is tab.
Syntax: $ paste file1 file2

tr

tr means translate.
This command translates character by character.
$ cat > demo.txt
While learning unix not required to eat.
1) $ tr 'aeiou' 'AEIOU' < demo.txt
It will replace lower case vowels with upper case vowels in demo.txt
WhIlE lEArnIng UnIx nOt rEqUIrEd tO EAt.

2) $ tr '[a-z]' '[A-Z]' < demo.txt
Every lower case alphabet symbol will be replaced with upper case alphabet symbol.
WHILE LEARNING UNIX NOT REQUIRED TO EAT

3) $ tr '[a-zA-Z]' '[A-Za-z]' < demo.txt
Every lower case character will be replaced with upper case character and every upper case character will be replaced with lower case character.
tr '[a-zA-Z]' '[A-Za-z]' < demo.txt > temp.txt

4) $ tr 'aeiou' '7' < demo.txt
To replace every lower case vowel with digit 7.

5) $ tr '|' '\t' < emp.dat
Replace | symbol with tab in emp.dat
$ tr '|' ':' < emp.dat

6) $ tr -d 'a' < demo.txt
-d means delete
It will delete all occurrence of 'a' in demo.txt
$ tr -d 'aeiou' < demo.txt
It will delete all lower case vowels in demo.txt

7) $ tr -s 'a' < demo.txt
It replaces sequence of a's with a single a.
-s means squeeze-repeats

Awk

Awk is a scripting language used for manipulating data and generating reports. The awk command programming language requires no compiling and allows the user to use variables, numeric functions, string functions, and logical operators.

Awk is a utility that enables a programmer to write tiny but effective programs in the form of statements that define text patterns that are to be searched for in each line of a document and the action that is to be taken when a match is found within a line. Awk is mostly used for pattern scanning and processing. It searches one or more files to see if they contain lines that match with the specified patterns and then perform the associated actions.

We can use awk commands to search in the files and print results in our required format.
AWK provides a more convenient way for file processing than our regular linux commands.
Most commonly used in shell scripting if we want to handle very large files.

WHAT CAN WE DO WITH AWK?

1. AWK Operations:
(a) Scans a file line by line
(b) Splits each input line into fields
(c) Compares input line/fields to pattern
(d) Performs action(s) on matched lines

2. Useful For:
(a) Transform data files
(b) Produce formatted reports

3. Programming Constructs:
(a) Format output lines
(b) Arithmetic and string operations
(c) Conditionals and loops

Structure of awk Command:

1) awk [option] {BEGIN BLOCK}{ACTION BLOCK}{END BLOCK} filenames
2) awk [options] 'selection _criteria {action }' input-file > output-file

BEGIN block will be executed only once before processing the file.
ACTION block will be executed for every line/record present in the file.
END block will be executed only once after completing all lines/records processing.

Options:
F → input Field Separator
f → takes blocks from a file

Predefined Variables of awk:
FS → Field Separator. Default FS is ' ' (space).
NR → Row Number / Number of Rows
NF → Number of Fields
RS → Record Separator
$N → Represents the column.
OFS → OFS command stores the output field separator, which separates the fields when Awk prints them. The default is a blank space. Whenever print has several parameters separated with commas, it will print the value of OFS in between each parameter.
ORS → ORS command stores the output record separator, which separates the output lines when Awk prints them. The default is a newline character. print automatically outputs the contents of ORS at the end of whatever it is given to print.

Examples:

1) awk '{print $1}' Afile1 →To print column 1.
2) awk '{print$1 "--->"$2}' Afile1 → To print col 1 → col 2.
3) awk '{print $0}' Afile1 → Print all columns.
4) awk -F '|' '{print $1 "--->" $2}' emp.dat → Now column separator is | instead of space.
5) awk '{print $1 "--->" $2}' FS='|' emp.dat → Now column separator is | instead of space.
6) awk 'NR != 1 {print $0}' Afile1 → All rows except First one.
7) awk 'NR == 1 {print $0}' Afile1 → Only First row.
8) awk 'NR > 2 {print $0}' Afile1 → All rows except First one and second.
9) awk 'NR == 2 || Nr == 4 {print $0}' Afile1 → Second and Fourth Row.
10) awk 'NR >= 2 && Nr <= 4 {print $0}' Afile1 → Second till Fourth Row.
11) awk 'NR == 2 , Nr == 4 {print $0}' Afile1 → Second till Fourth Row.
12) awk '$3>20 {print $0}' Afile1 → Rows whose 3rd col value is greater than 20.
13) awk '{print}' Afile1 → Print whole file.
14) awk '/s/{print $0}' Afile1 → Search pattern s and print that line.
15) awk '{if($3 > 30){$3=$3+800; print $0}}' → If using awk.
16) awk '{print NR " " $0} → Print row Numbers
17) awk 'END{print NR}' → Print total Row Numbers.
18) awk 'BEGIN{for(i=1;i<=10;i++) print "square of ",i," is ",i*i}' →Print square
19) awk 'BEGIN{OFS="---->"} {print $1,$2,$3}' Afile1 → OFS example.
20) awk 'BEGIN{ORS="::::"} {print $1}' Afile1 → ORS example.
21) awk -f script Afile1 → Take script from file.
22) awk 'BEGIN{c=0} NR != 1{c = c+$2} END{print "Total age: ",c} Afile1 → Prints total Age.
23) awk '{ if (length($0) > max) max = length($0) } END { print max }' Afile1 → Max length of a row.
24) awk 'length($0) > 10' Afile1 → Printing lines with more than 10 characters

MCQs

Awk

Certainly! Here are 30 challenging multiple-choice questions related to the `awk` command along with their answers:

1. What does the following `awk` command do?
  ```bash
  awk '{print $2, $1}' file.txt
  ```
  A) Prints the first and second columns in reverse order for each line in file.txt
  B) Prints the second and first columns in reverse order for each line in file.txt
  C) Prints the first and second columns separated by a comma for each line in file.txt
  D) Prints the second column followed by the first column for each line in file.txt
  Answer: B) Prints the second and first columns in reverse order for each line in file.txt

2. Which `awk` command prints the number of fields for each line in a file 'data.txt'?
  A) `awk '{print NF}' data.txt`
  B) `awk 'NF {print NF}' data.txt`
  C) `awk '/./ {print NF}' data.txt`
  D) `awk '/\S+/ {print NF}' data.txt`
  Answer: A) `awk '{print NF}' data.txt`

3. What does the following `awk` command do?
  ```bash
  awk -F':' '$3 >= 1000 && $3 <= 2000 {print $1}' /etc/passwd
  ```
  A) Prints the third field if it falls between 1000 and 2000 separated by a colon in /etc/passwd
  B) Prints the first field if the third field falls between 1000 and 2000 separated by a colon in /etc/passwd
  C) Prints the lines where the third field falls between 1000 and 2000 separated by a colon in /etc/passwd
  D) Prints the lines that contain 1000 or 2000 in the third field separated by a colon in /etc/passwd

Answer: B) Prints the first field if the third field falls between 1000 and 2000 separated by a colon in /etc/passwd

4. Which `awk` command prints lines from a file 'file.txt' where the length of the first field is greater than 5?
A) `awk 'length($1) > 5' file.txt`
B) `awk '{if (length($1) > 5) print $0}' file.txt`
C) `awk '$1 ~ /^.{6,}$/' file.txt`
D) `awk '/^.{6,}/' file.txt`
Answer: A) `awk 'length($1) > 5' file.txt`

5. What does the following `awk` command do?
```bash
awk '/pattern/ {for(i=NF;i>0;i--) printf("%s ",$i); print ""}' file.txt
```
A) Prints lines in reverse order for each line containing 'pattern' in file.txt
B) Prints lines containing 'pattern' in reverse order in file.txt
C) Prints fields in reverse order for each line containing 'pattern' in file.txt
D) Prints fields containing 'pattern' in reverse order in file.txt
Answer: C) Prints fields in reverse order for each line containing 'pattern' in file.txt

6. Which `awk` command prints lines where the sum of the second and third columns is greater than 100 in a file 'data.txt'?
A) `awk '$2 + $3 > 100' data.txt`
B) `awk '{if ($2 + $3 > 100) print $0}' data.txt`
C) `awk '$2 + $3 > 100 {print $1}' data.txt`
D) `awk '{if ($2 + $3 > 100) print $1}' data.txt`
Answer: A) `awk '$2 + $3 > 100' data.txt`

7. What does the following `awk` command do?
```bash
awk -F':' '{ if ($3 > 10) print $1 }' /etc/passwd
```
A) Prints lines from /etc/passwd where the first field is greater than 10
B) Prints lines from /etc/passwd where the third field is greater than 10
C) Prints lines from /etc/passwd where the first field is greater than ':'
D) Prints lines from /etc/passwd where the third field is greater than ':'
Answer: B) Prints lines from /etc/passwd where the third field is greater than 10

8. Which `awk` command prints lines from a file 'info.txt' where the third column contains the string 'keyword'?
A) `awk '$3 ~ /keyword/' info.txt`
B) `awk '$3 == /keyword/' info.txt`
C) `awk '$3 == "keyword"' info.txt`

D) `awk '$3 ~ "keyword"' info.txt`
Answer: A) `awk '$3 ~ /keyword/' info.txt`

9. What does the following `awk` command do?
```bash
awk '{ total += $2 } END { print total }' data.txt
```
A) Prints the sum of the second column in data.txt
B) Prints the total number of columns in data.txt
C) Prints the sum of all columns in data.txt
D) Prints the average of the second column in data.txt
Answer: A) Prints the sum of the second column in data.txt

10. Which `awk` command prints lines where the length of the second field is less than 4 characters in a file 'file.txt'?
A) `awk 'length($2) < 4' file.txt`
B) `awk '{if (length($2) < 4) print $0}' file.txt`
C) `awk '$2 ~ /^.{1,3}$/' file.txt`
D) `awk '/^.{1,3}$/' file.txt`
Answer: A) `awk 'length($2) < 4' file.txt`

11. What does the following `awk` command do?
```bash
awk '$1 ~ /regex/ { count++ } END { print count }' file.txt
```
A) Counts occurrences of 'regex' in the first field of each line in file.txt
B) Counts lines that match the regular expression 'regex' in file.txt
C) Counts lines where the first field matches the regular expression 'regex' in file.txt
D) Prints the number of occurrences of 'regex' in file.txt
Answer: C) Counts lines where the first field matches the regular expression 'regex' in file.txt

12. Which `awk` command prints the second and fourth fields separated by a comma for each line in a file 'data.txt'?
A) `awk '{print $2, ",", $4}' data.txt`
B) `

awk '{print $2, $4}' FS=, OFS=, data.txt`
C) `awk -F',' '{print $2 ", " $4}' data.txt`
D) `awk '$2, $4 {print FS=","}' data.txt`
Answer: C) `awk -F',' '{print $2 ", " $4}' data.txt`

13. What does the following `awk` command do?
```bash
```

```
awk 'length($0) > max_length { max_length = length($0); longest_line = $0 } END { print longest_line }' file.txt
```

A) Prints the longest line in file.txt
B) Prints the number of characters in the longest line in file.txt
C) Prints the number of lines longer than max_length in file.txt
D) Prints lines that are longer than max_length in file.txt
Answer: A) Prints the longest line in file.txt

14. Which `awk` command prints lines that contain both 'word1' and 'word2' in any order in a file 'text.txt'?
A) `awk '/word1/ && /word2/' text.txt`
B) `awk '/word1/ || /word2/' text.txt`
C) `awk '/word1/ && $0 ~ /word2/' text.txt`
D) `awk '/word1/ || $0 ~ /word2/' text.txt`
Answer: A) `awk '/word1/ && /word2/' text.txt`

15. What does the following `awk` command do?
```bash
awk '/pattern/ {print FNR, $0}' file.txt
```
A) Prints line numbers and lines containing 'pattern' in file.txt
B) Prints line numbers where 'pattern' occurs in file.txt
C) Prints lines containing 'pattern' along with their line numbers in file.txt
D) Prints line numbers and lines that do not contain 'pattern' in file.txt
Answer: A) Prints line numbers and lines containing 'pattern' in file.txt

16. Which `awk` command is used to calculate the average of the second column in a file 'data.txt'?
A) `awk '{ total += $2; count++ } END { print total / count }' data.txt`
B) `awk '{ sum += $2 } END { print sum / NR }' data.txt`
C) `awk '{ total += $2 } END { print total / NR }' data.txt`
D) `awk '{ if ($2 > 0) sum += $2 } END { print sum / NR }' data.txt`
Answer: C) `awk '{ total += $2 } END { print total / NR }' data.txt`

17. What does the following `awk` command do?
```bash
awk -F'|' 'NF != 3' file.txt
```
A) Prints lines where the number of fields is not equal to 3 separated by '|' in file.txt
B) Prints lines where the number of fields is equal to 3 separated by '|' in file.txt
C) Deletes lines where the number of fields is not equal to 3 separated by '|' in file.txt
D) Deletes lines where the number of fields is equal to 3 separated by '|' in file.txt

Answer: A) Prints lines where the number of fields is not equal to 3 separated by '|' in file.txt

18. Which `awk` command is used to count the number of lines starting with a specific word 'start' in a file 'file.txt'?
A) `awk '$1 ~ /^start/' file.txt`
B) `awk '/^start/ { count++ } END { print count }' file.txt`
C) `awk '$0 ~ /^start/ { count++ } END { print count }' file.txt`
D) `awk '/^start/ {print NR}' file.txt`
Answer: B) `awk '/^start/ { count++ } END { print count }' file.txt`

19. What does the following `awk` command do?
```bash
awk '{ for (i = NF; i > 0; i--) printf("%s ", $i); print "" }' file.txt
```
A) Prints fields in reverse order for each line in file.txt
B) Prints lines in reverse order in file.txt
C) Reverses the order of fields and lines in file.txt
D) Reverses the order of characters in each field in file.txt
Answer: A) Prints fields in reverse order for each line in file.txt

20. Which `awk` command is used to print the number of occurrences of a specific word 'count' in a file 'file.txt'?
A) `awk '/count/ {print NR}' file.txt`
B) `awk '/count/ { count++ } END { print count }' file.txt`
C) `awk '$0 ~ /count/ { count++ } END { print count }' file.txt`
D) `awk '/count/ {print $0}' file.txt`
Answer: B) `awk '/count/ { count++ } END { print count }' file.txt`

21. What does the following `awk` command do?
```bash
awk '/search/{gsub(/old/,"new")}1' file.txt
```
A) Searches for lines containing 'search' and replaces 'old' with 'new' in file.txt
B) Searches for lines containing 'old' and replaces it with 'new' in file.txt
C) Searches for lines containing 'search' and deletes 'old' in file.txt
D) Searches for lines containing 'old' and appends 'new' to them in file.txt
Answer: A) Searches for lines containing 'search' and replaces 'old' with 'new' in file.txt

Absolutely! Here are ten challenging multiple-choice questions related to the `awk` command along with their answers:

1. What does the following `awk` command do?
```bash
awk '{ if ($1 > max) max = $1 } END { print max }' file.txt
```
A) Finds the sum of the first column in file.txt
B) Finds the maximum value in the first column of file.txt
C) Prints the last value in the first column of file.txt
D) Prints the number of lines in file.txt
Answer: B) Finds the maximum value in the first column of file.txt

2. Which `awk` command prints lines from a file.txt where the second column contains the word 'search'?
A) `awk '/search/' file.txt`
B) `awk '$2 == "search"' file.txt`
C) `awk '$2 ~ /search/' file.txt`
D) `awk '$2 contains "search"' file.txt`
Answer: B) `awk '$2 == "search"' file.txt`

3. What does the following `awk` command do?
```bash
awk '{ sum += $NF } END { print sum/NR }' file.txt
```
A) Calculates the average of the last column in file.txt
B) Calculates the sum of all columns in file.txt
C) Finds the average value in the first column of file.txt
D) Calculates the total number of lines in file.txt
Answer: A) Calculates the average of the last column in file.txt

4. Which `awk` command prints lines from a file.txt where the length of the third column is greater than 5 characters?
A) `awk 'length($3) > 5' file.txt`
B) `awk '$3 >= 5' file.txt`
C) `awk '$3 ~ /^.{5,}$/' file.txt`
D) `awk '/.{5,}/' file.txt`

Answer: A) `awk 'length($3) > 5' file.txt`

5. What does the following `awk` command do?
```bash
awk '/pattern/ {count++} END {print count}' file.txt
```
A) Counts the number of lines in file.txt containing 'pattern'
B) Counts the occurrences of 'pattern' in each line of file.txt
C) Prints the number of lines in file.txt containing 'pattern'
D) Prints the total count of 'pattern' in file.txt
Answer: A) Counts the number of lines in file.txt containing 'pattern'

6. Which `awk` command prints lines from a file.txt where the sum of the first and second columns is greater than 50?
A) `awk '$1 + $2 > 50' file.txt`
B) `awk '$1 && $2 > 50' file.txt`
C) `awk '$1 * $2 > 50' file.txt`
D) `awk '$1 > 50 && $2 > 50' file.txt`
Answer: A) `awk '$1 + $2 > 50' file.txt`

7. What does the following `awk` command do?
```bash
awk 'BEGIN {FS = ":"} {print $3}' /etc/passwd
```
A) Prints the first three fields separated by a colon in /etc/passwd
B) Prints the third field separated by a colon in /etc/passwd
C) Sets the field separator as a colon and prints all fields in /etc/passwd
D) Prints the last three fields separated by a colon in /etc/passwd
Answer: B) Prints the third field separated by a colon in /etc/passwd

8. Which `awk` command is used to print only the lines of file.txt that are longer than 20 characters?
A) `awk 'length > 20' file.txt`
B) `awk '/^.{20,}$/' file.txt`
C) `awk 'length($0) > 20' file.txt`
D) `awk 'length($0) ~ /^.{20,}$/' file.txt`
Answer: C) `awk 'length($0) > 20' file.txt`

9. What does the following `awk` command do?
```bash
awk 'NR % 2 == 0' file.txt
```
A) Prints every second line starting from the first line in file.txt
B) Prints the even-numbered lines in file.txt

C) Deletes every second line in file.txt
D) Skips every second line while processing file.txt
Answer: B) Prints the even-numbered lines in file.txt

10. Which `awk` command prints lines from a file.txt where the sum of all columns is greater than 100?
A) `awk 'sum($0) > 100' file.txt`
B) `awk '{sum += $0} END {if (sum > 100) print $0}' file.txt`
C) `awk '{sum = 0} {for (i = 1; i <= NF; i++) sum += $i} {if (sum > 100) print $0}' file.txt`
D) `awk 'sum($NF) > 100' file.txt`
Answer: C) `awk '{sum = 0} {for (i = 1; i <= NF; i++) sum += $i} {if (sum > 100) print $0}' file.txt`

Sed

Absolutely! Here are 20 challenging multiple-choice questions related to the `sed` command along with their answers:

1. What does the following `sed` command do?
```bash
sed 's/\(a\+\) \(b\+\)/\2 \1/g' file.txt
```
A) Swaps the order of 'a' and 'b' in each line of file.txt
B) Replaces 'a' with 'b' and 'b' with 'a' in file.txt
C) Reverses the order of 'a' and 'b' in each line of file.txt
D) Replaces 'ab' with 'ba' in file.txt
Answer: C) Reverses the order of 'a' and 'b' in each line of file.txt

2. Which `sed` command is used to delete lines containing 'pattern' and the next two lines in a file?
A) `sed '/pattern/,+2d' file.txt`
B) `sed '/pattern/{N;N;d;}' file.txt`
C) `sed '/pattern/,2d' file.txt`
D) `sed '/pattern/,+2p' file.txt`
Answer: A) `sed '/pattern/,+2d' file.txt`

3. What does the following `sed` command do?
```bash
```

sed 's/\([0-9]\{3\}\)/-\1-/g' file.txt
```

A) Replaces all occurrences of three consecutive digits with hyphens around them in file.txt
B) Adds hyphens before and after every digit in file.txt
C) Deletes every third digit in file.txt
D) Replaces all occurrences of one or more digits with hyphens in file.txt
**Answer: A) Replaces all occurrences of three consecutive digits with hyphens around them in file.txt**

**4. Which `sed` command is used to append 'New Line' after every 5 lines in a file named 'data.txt'?**
A) `sed '5a New Line' data.txt`
B) `sed '5i New Line' data.txt`
C) `sed '5~5a New Line' data.txt`
D) `sed '5~5i New Line' data.txt`
**Answer: C) `sed '5~5a New Line' data.txt`**

**5. What does the following `sed` command do?**
```bash
sed -n '1~3p' file.txt
```

A) Prints the first line of every third block of three lines in file.txt
B) Prints the first three lines of every block of three lines in file.txt
C) Deletes every third line in file.txt
D) Replaces every third line in file.txt with the first line
Answer: A) Prints the first line of every third block of three lines in file.txt

6. Which `sed` command deletes all trailing whitespace characters from each line in a file?
A) `sed 's/\s*$//' file.txt`
B) `sed 's/\s*$//g' file.txt`
C) `sed 's/\s*//' file.txt`
D) `sed 's/\s+/g' file.txt`
Answer: B) `sed 's/\s*$//g' file.txt`

7. What does the following `sed` command do?
```bash
sed '/^$/d' file.txt
```

A) Deletes empty lines from file.txt
B) Deletes lines starting with a dollar sign ($) from file.txt
C) Deletes lines ending with a dollar sign ($) from file.txt
D) Deletes lines starting with a caret (^) from file.txt
Answer: A) Deletes empty lines from file.txt
```

8. Which `sed` command adds a line 'Inserted' after the last line in a file 'info.txt'?
A) `sed '$i Inserted' info.txt`
B) `sed '$a Inserted' info.txt`
C) `sed '/$$/a Inserted' info.txt`
D) `sed '/$$/{a\Inserted}' info.txt`
Answer: B) `sed '$a Inserted' info.txt`

9. What does the following `sed` command do?
```bash
sed -e '/^$/d' -e 's/\(.*\)/\U\1/' file.txt
```
A) Deletes empty lines and converts all text to uppercase in file.txt
B) Converts all text to uppercase and then deletes empty lines in file.txt
C) Deletes empty lines and converts only the first character of each line to uppercase in file.txt
D) Converts all text to uppercase and appends a line with the content 'file.txt'
Answer: A) Deletes empty lines and converts all text to uppercase in file.txt

10. Which `sed` command substitutes the third occurrence of a word 'find' with 'replace' in a line?
A) `sed 's/find/replace/3'`
B) `sed 's/find/replace/g3'`
C) `sed 's/find/replace/3g'`
D) `sed 's/find/replace/g3'`
Answer: A) `sed 's/find/replace/3'`

11. What does the following `sed` command do?
```bash
sed '/pattern/{x;s/^/prefix-/;x}' file.txt
```
A) Adds 'prefix-' at the beginning of lines containing 'pattern' in file.txt
B) Replaces 'pattern' with 'prefix-' in file.txt
C) Appends 'prefix-' at the end of lines containing 'pattern' in file.txt
D) Deletes lines containing 'pattern' in file.txt
Answer: A) Adds 'prefix-' at the beginning of lines containing 'pattern' in file.txt

12. Which `sed` command substitutes the second occurrence of a pattern 'abc' with 'xyz' in a line?
A) `sed 's/abc/xyz/2'`
B) `sed 's/abc/xyz/g2'`
C) `sed 's/abc/xyz/2g'`
D) `sed 's/abc/xyz/g2'`
Answer: A) `sed 's/abc/xyz/2'`

13. What does the following `sed` command do?

```bash
sed '1~2d' file.txt
```

A) Deletes every second line starting from the first line in file.txt
B) Deletes the first two lines in file.txt
C) Deletes lines containing the character '2' in file.txt
D) Deletes lines between the first and second line in file.txt

Answer: A) Deletes every second line starting from the first line in file.txt

14. Which `sed` command inserts a line 'New Text' before the line containing the pattern 'pattern' in a file?
A) `sed '/pattern/i New Text' file.txt`
B) `sed '/pattern/a New Text' file.txt`
C) `sed '/pattern/c New Text' file.txt`
D) `sed '/pattern/r New Text' file.txt`
Answer: A) `sed '/pattern/i New Text' file.txt`

15. What does the following `sed` command do?
```bash
sed 's/old/new/gw newfile.txt' file.txt
```

A) Replaces 'old' with 'new' globally in file.txt and writes the output to newfile.txt
B) Replaces 'old' with 'new' globally in file.txt and appends the output to newfile.txt
C) Writes the output of replacing 'old' with 'new' globally in file.txt to newfile.txt
D) Replaces 'old' with 'new' globally in file.txt and saves the original file as newfile.txt
Answer: C) Writes the output of replacing 'old' with 'new' globally in file.txt to newfile.txt

16. Which `sed` command is used to delete all occurrences of a specific word 'delete' in a file 'data.txt'?
A) `sed '/delete/d' data.txt`
B) `sed 's/delete//g' data.txt`
C) `sed 's/delete//gw output.txt' data.txt`
D) `sed '/delete/s///g' data.txt`
Answer: B) `sed 's/delete//g' data.txt`

17. What does the following `sed` command do?
```bash
sed -e '1d' -e '$d' file.txt
```

A) Deletes the first and last lines in file.txt
B) Deletes the second and penultimate lines in file.txt
C) Deletes lines starting with a digit in file.txt

D) Deletes lines ending with a dollar sign ($) in file.txt
Answer: A) Deletes the first and last lines in file.txt

18. Which `sed` command prints only lines that do not contain the word 'error' from a file 'log.txt'?
A) `sed '/error/p' log.txt`
B) `sed '/error/!p' log.txt`
C) `sed '/error/d' log.txt`
D) `sed '/!error/p' log.txt`
Answer: B) `sed '/error/!p' log.txt`

19. What does the following `sed` command do?
```bash
sed 's/^/# /g' file.txt > newfile.txt
```
A) Adds a hash symbol at the beginning of each line in file.txt and saves the output in newfile.txt
B) Deletes all lines starting with a hash symbol in file.txt and saves the output in newfile.txt
C) Appends a line with the content '# file.txt' at the end of newfile.txt
D) Adds a hash symbol at the end of each line in file.txt and saves the output in newfile.txt
Answer: A) Adds a hash symbol at the beginning of each line in file.txt and saves the output in newfile.txt

20. Which `sed` command substitutes the second to fourth occurrences of a pattern 'pattern' with 'replace' in a line?
A) `sed 's/pattern/replace/2,4'`
B) `sed 's/pattern/replace/2;4'`
C) `sed 's/pattern/replace/2g;4g'`
D) `sed 's/pattern/replace/2g;4'`
Answer: A) `sed 's/pattern/replace/2,4'`

These questions delve into various functionalities and nuances of the `sed` command in Unix/Linux systems, covering pattern matching, substitution, deletion, line insertion, and other key operations. Understanding these concepts and commands is crucial for effective text processing using `sed`.

Certainly! Here are ten challenging multiple-choice questions related to the `sed` command along with their answers:

1. What does the following `sed` command do?
```bash
```

sed 's/old/new/g' file.txt
```
A) Replaces the first occurrence of 'old' with 'new' in file.txt
B) Replaces all occurrences of 'old' with 'new' in file.txt
C) Replaces 'new' with 'old' in file.txt
D) Deletes 'old' from file.txt
**Answer: B) Replaces all occurrences of 'old' with 'new' in file.txt**

**2. Which `sed` command would delete all blank lines from a file?**
A) `sed '/^$/d' file.txt`
B) `sed '/^\s*$/d' file.txt`
C) `sed '/\n/d' file.txt`
D) `sed '/\t/d' file.txt`
**Answer: A) `sed '/^$/d' file.txt`**

**3. What does the following `sed` command do?**
```bash
sed '1,3s/old/new/g' file.txt
```
A) Replaces 'old' with 'new' in lines 1 to 3 of file.txt
B) Replaces 'old' with 'new' in lines 1 and 3 of file.txt
C) Replaces 'old' with 'new' in line 3 of file.txt
D) Replaces 'old' with 'new' in lines starting from line 1 to line 3 in file.txt
Answer: A) Replaces 'old' with 'new' in lines 1 to 3 of file.txt

4. Which `sed` command is used to append a line after the last line of a file?
A) `sed '$a text to append' file.txt`
B) `sed '/$/{a\text to append}' file.txt`
C) `sed '$i text to append' file.txt`
D) `sed '/$$/a text to append' file.txt`
Answer: A) `sed '$a text to append' file.txt`

5. What does the following `sed` command do?
```bash
sed '/pattern/{n;s/word/replace/}' file.txt
```
A) Replaces 'word' with 'replace' on lines containing 'pattern' in file.txt
B) Deletes the line following 'pattern' if it contains 'word' in file.txt
C) Skips the line containing 'pattern' and replaces 'word' with 'replace' in the next line
D) Replaces 'word' with 'replace' in the line following 'pattern' in file.txt
Answer: D) Replaces 'word' with 'replace' in the line following 'pattern' in file.txt

6. Which `sed` command substitutes the third occurrence of a word 'find' with 'replace' in a line?
```

A) `sed 's/find/replace/3'`
B) `sed 's/find/replace/g3'`
C) `sed 's/find/replace/3g'`
D) `sed 's/find/replace/g3'`
Answer: A) `sed 's/find/replace/3'`

7. What is the purpose of the following `sed` command?
```bash
sed '/pattern/{x;s/^/prefix-/;x}' file.txt
```
A) Adds 'prefix-' at the beginning of lines containing 'pattern' in file.txt
B) Replaces 'pattern' with 'prefix-' in file.txt
C) Appends 'prefix-' at the end of lines containing 'pattern' in file.txt
D) Deletes lines containing 'pattern' in file.txt
Answer: A) Adds 'prefix-' at the beginning of lines containing 'pattern' in file.txt

8. Which `sed` command is used to delete the last character of each line in a file?
A) `sed 's/.$//' file.txt`
B) `sed 's/$/d' file.txt`
C) `sed 's/^.//' file.txt`
D) `sed 's/.$/d' file.txt`
Answer: A) `sed 's/.$//' file.txt`

9. What does the following `sed` command do?
```bash
sed '1,5!d' file.txt
```
A) Deletes lines 1 to 5 in file.txt
B) Prints lines 1 to 5 in file.txt
C) Deletes all lines except lines 1 to 5 in file.txt
D) Deletes lines that contain '1' or '5' in file.txt
Answer: B) Prints lines 1 to 5 in file.txt

10. Which `sed` command is used to replace the second occurrence of a pattern 'abc' with 'xyz' in a line?
A) `sed 's/abc/xyz/2'`
B) `sed 's/abc/xyz/g2'`
C) `sed 's/abc/xyz/2g'`
D) `sed 's/abc/xyz/g2'`
Answer: A) `sed 's/abc/xyz/2'`

These questions cover various functionalities and usage scenarios of the `sed` command in Unix/Linux systems and might require a good understanding of its basic operations and syntax to answer correctly.