

Practice Questions

Topic - “Regular expressions, redirections: I/P, O/P, Error.”

Multiple-Choice Questions (MCQs) –

1. What is the purpose of regular expressions in Linux?

- A) To design graphical user interfaces
- B) To search, match, and manipulate text patterns
- C) To manage hardware drivers
- D) To create database schemas

Answer: B

2. Which Linux command uses regular expressions to search for patterns in text?

- A) ls
- B) grep
- C) cd
- D) mkdir

Answer: B

3. What does the > operator do in Linux redirections?

- A) Redirects standard input
- B) Redirects standard output to a file (overwrites)
- C) Redirects standard error
- D) Appends standard output to a file

Answer: B

4. What is the purpose of the >> operator in Linux?

- A) Redirects standard input
- B) Redirects standard output to a file (appends)
- C) Redirects standard error
- D) Overwrites standard output

Answer: B

Practice Questions

5. Which regular expression metacharacter matches the start of a line?

- A) \$
- B) ^
- C) *
- D) .

Answer: B

6. What does the 2> operator do in Linux redirections?

- A) Redirects standard input
- B) Redirects standard output
- C) Redirects standard error to a file
- D) Appends standard output to a file

Answer: C

7. Which regular expression metacharacter matches any single character?

- A) *
- B) .
- C) +
- D) ?

Answer: B

8. What is the purpose of the < operator in Linux?

- A) Redirects standard output
- B) Redirects standard input from a file
- C) Redirects standard error
- D) Appends standard output to a file

Answer: B

Practice Questions

9. Which command is used to redirect both standard output and standard error to the same file?

- A) command > file
- B) command 2> file
- C) command &> file
- D) command < file

Answer: C

10. What does the regular expression `a*` match?

- A) One or more a characters
- B) Zero or more a characters
- C) Exactly one a character
- D) Zero or one a character

Answer: B

11. Which regular expression metacharacter matches the end of a line?

- A) ^
- B) \$
- C) *
- D) .

Answer: B

12. What is the purpose of the `|` operator in Linux?

- A) Redirects standard input
- B) Pipes the output of one command as input to another
- C) Redirects standard error
- D) Appends standard output to a file

Answer: B

Practice Questions

13. Which regular expression matches a digit in Linux?

- A) \d
- B) [0-9]
- C) \w
- D) \s

Answer: B

14. What does the command `ls > output.txt` do in Linux?

- A) Redirects the input of `ls` to `output.txt`
- B) Redirects the output of `ls` to `output.txt` (overwrites)
- C) Redirects the error of `ls` to `output.txt`
- D) Appends the output of `ls` to `output.txt`

Answer: B

15. Which regular expression matches one or more occurrences of a character?

- A) *
- B) +
- C) ?
- D) .

Answer: B

16. What does the command `sort < input.txt > output.txt` do in Linux?

- A) Redirects the output of `sort` to `input.txt`
- B) Takes input from `input.txt`, sorts it, and writes the output to `output.txt`
- C) Redirects the error of `sort` to `output.txt`
- D) Appends the output of `sort` to `input.txt`

Answer: B

Practice Questions

17. Which regular expression matches whitespace characters?

- A) \d
- B) \w
- C) \s
- D) \t

Answer: C

18. What does the command `command 2>&1 > output.txt` do in Linux?

- A) Redirects standard input to output.txt
- B) Redirects standard output and standard error to output.txt
- C) Redirects standard input and standard error to output.txt
- D) Appends standard output to output.txt

Answer: B

19. Which regular expression matches zero or one occurrence of a character?

- A) *
- B) +
- C) ?
- D) .

Answer: C

20. What is the purpose of the `tee` command in Linux redirections?

- A) To redirect standard input
- B) To redirect output to both a file and the terminal
- C) To redirect standard error
- D) To append standard output to a file

Answer: B

Practice Questions

Short Descriptive Questions –

1. Explain the purpose of regular expressions in Linux and provide two examples of their use with the grep command.
2. Describe the difference between the > and >> operators in Linux redirections, with examples.
3. Discuss the significance of the | (pipe) operator in Linux and provide an example of its use.
4. Explain the role of the 2> operator in redirecting standard error in Linux, with an example.
5. Describe the use of the ^ and \$ metacharacters in regular expressions, with examples.
6. Explain how the < operator is used to redirect standard input in Linux, with an example.
7. Discuss the importance of using the &> operator for redirecting both standard output and standard error, with an example.
8. Describe the use of the *, +, and ? metacharacters in regular expressions, with examples.
9. Explain how the tee command can be used in Linux redirections, with an example.
10. Discuss the role of character classes (e.g., [0-9], [a-z]) in regular expressions, with examples.

Practice Questions

Long Descriptive Questions –

1. Compare and contrast the use of regular expressions in Linux commands like grep, sed, and awk, highlighting their strengths and weaknesses in text pattern matching and manipulation.
2. Discuss the steps involved in using redirections in Linux to manage input, output, and error streams, emphasizing the role of operators like >, >>, <, 2>, and |. Provide examples for each.
3. Analyze the challenges users face in working with regular expressions in Linux, such as complexity, escaping special characters, and debugging, and suggest strategies to overcome these challenges.
4. Evaluate the importance of redirections in Linux for scripting and automation, focusing on scenarios like logging errors, piping data between commands, and saving outputs. Provide a step-by-step plan for creating a script that uses redirections to process and log data.
5. Explain the concept of regular expressions in Linux and discuss how they can be used to solve real-world problems, such as log file analysis, data extraction, and text replacement. Provide a step-by-step plan for using grep and sed with regular expressions to extract and modify data from a log file.

THE END