**LAB – ASSIGNMENT – 02  
COMMANDS**

**Part 1: Outputs of Commands**

To provide accurate outputs, I'll demonstrate the commands and explain their typical results. Remember, the actual output will depend on your specific file system and data.

1. **pwd** (Print Working Directory):
   * Output: /home/yourusername (or the current directory path)
   * Purpose: Displays the absolute path of the current directory.
2. **cd** (Change Directory):
   * Example: cd Documents
   * Output: (No visible output, but the working directory changes)
   * Purpose: Changes the current directory.
3. **ls** (List):
   * Output: (Lists files and directories in the current directory)
   * Example: ls
   * Example with flags: ls -lha (lists detailed info, human-readable sizes, all files)
   * Purpose: Lists directory contents.
4. **mkdir** (Make Directory):
   * Example: mkdir mydirectory
   * Output: (No visible output, but the directory is created)
   * Purpose: Creates a new directory.
5. **rm** (Remove):
   * Example: rm myfile.txt
   * Output: (No visible output, but the file is deleted)
   * Purpose: Removes files or directories. (Use -r for directories)
6. **touch** (Touch):
   * Example: touch newfile.txt
   * Output: (No visible output, but the file is created or its timestamp updated)
   * Purpose: Creates an empty file or updates a file's timestamp.
7. **hostname** (Hostname):
   * Output: yourcomputername
   * Purpose: Displays the system's hostname.
8. **cat** (Concatenate):
   * Example: cat myfile.txt
   * Output: (Displays the content of myfile.txt)
   * Purpose: Displays file contents, concatenates files.
9. **chmod** (Change Mode):
   * Example: chmod 755 script.sh
   * Output: (No visible output, but file permissions are changed)
   * Purpose: Changes file permissions.
10. **echo** (Echo):
    * Example: echo "Hello, world!"
    * Output: Hello, world!
    * Purpose: Displays text.
11. **grep** (Global Regular Expression Print):
    * Example: grep "pattern" myfile.txt
    * Output: (Lines containing "pattern" from myfile.txt)
    * Purpose: Searches for patterns in files.
12. **fgrep** (Fixed grep):
    * Example: fgrep "literal string" myfile.txt
    * Output: (Lines containing "literal string" from myfile.txt)
    * Purpose: Searches for fixed strings (faster than grep for simple strings).
13. **mv** (Move):
    * Example: mv oldfile.txt newfile.txt
    * Output: (No visible output, but the file is renamed or moved)
    * Purpose: Moves or renames files/directories.
14. **cp** (Copy):
    * Example: cp myfile.txt copy.txt
    * Output: (No visible output, but the file is copied)
    * Purpose: Copies files/directories.
15. **more** (More):
    * Example: more largefile.txt
    * Output: (Displays largefile.txt one screen at a time)
    * Purpose: Displays file content one screen at a time.
16. **less** (Less):
    * Example: less largefile.txt
    * Output: (Displays largefile.txt one screen at a time, with more navigation options)
    * Purpose: Similar to more, but more versatile.
17. **wc** (Word Count):
    * Example: wc myfile.txt
    * Output: (Lines, words, bytes count)
    * Purpose: Counts lines, words, and bytes in a file.
18. **awk** (AWK):
    * Example: awk '{print $1}' myfile.txt
    * Output: (Prints the first column of each line)
    * Purpose: Text processing, data extraction.
19. **sed** (Stream Editor):
    * Example: sed 's/old/new/g' myfile.txt
    * Output: (Modified content of myfile.txt displayed)
    * Purpose: Text manipulation, search and replace.
20. **tail** (Tail):
    * Example: tail myfile.txt
    * Output: (Last 10 lines of myfile.txt)
    * Example: tail -n 20 myfile.txt (last 20 lines)
    * Purpose: Displays the end of a file.

**Part 2: Answers to Questions**

1. **How to navigate to a Specific Directory?**
   * Command: cd /path/to/directory (absolute path) or cd directoryname (relative path).
2. **How to see detailed information about files and directories using ls?**
   * Command: ls -lha
3. **How to create multiple directories in Linux using mkdir command?**
   * Command: mkdir dir1 dir2 dir3 or mkdir -p parentdir/subdir1/subdir2
4. **How to remove multiple files at once with rm?**
   * Command: rm file1.txt file2.txt file3.txt
5. **Can rm be used to delete directories?**
   * Yes, with the -r or -rf option: rm -r directoryname (or rm -rf directoryname for force delete).
6. **How Do You Copy Files and Directories in Linux?**
   * Files: cp sourcefile.txt destinationfile.txt
   * Directories: cp -r sourcedirectory destinationdirectory
7. **How to Rename a file in Linux Using mv Command?**
   * Command: mv oldfilename.txt newfilename.txt
8. **How to Move Multiple files in Linux Using mv Command?**
   * Command: mv file1.txt file2.txt destinationdirectory/
9. **How to Create Multiple Empty Files by Using Touch Command in Linux?**
   * Command: touch file1.txt file2.txt file3.txt
10. **How to View the Content of Multiple Files in Linux?**
    * Command: cat file1.txt file2.txt file3.txt
11. **How to Create a file and add content in Linux Using cat Command?**
    * Command: cat > filename.txt (then type content and press Ctrl+D)
12. **How to Append the Contents of One File to the End of Another File using cat command?**
    * Command: cat file1.txt >> file2.txt
13. **How to use cat command if the file has a lot of content and can’t fit in the terminal.**
    * Use less filename.txt or more filename.txt
14. **How to Merge Contents of Multiple Files Using cat Command?**
    * Command: cat file1.txt file2.txt > mergedfile.txt
15. **How to use cat Command to Append to an Existing File**
    * Command: cat >> file.txt (then type the content to append, and press Ctrl+D).
16. **What is “chmod 777 “, “chmod 755” and “chmod +x “or “chmod a+x”?**
    * chmod 777: Grants read, write, and execute permissions to the owner, group, and others.
    * chmod 755: Grants read, write, and execute permissions to the owner, read and execute permissions to the group and others.
    * chmod +x or chmod a+x: adds execute permissions to everyone.
17. **How to find the number of lines that matches the given string/pattern?**
    * Command: grep -c "pattern" filename.txt
18. **How to display the files that contains the given string/pattern.?**
    * Command: grep -l "pattern" \*
19. **How to show the line number of file with the line matched.?**
    * Command: `grep -n "