Chapter 3 Short Questions with Answers

Q1: How to Login to a UNIX system?

- Enter **username** at login prompt.
- Enter **password** when asked.
- System displays **welcome message & shell prompt**.
  - ☐ *Example:* login: ali → *Password:* **** → *$ prompt appears.*

Q2: How to Logout of a UNIX system?

- Use the command: logout (if using shell).
- Or press **Ctrl + D** to terminate the session.
  - ☐ *Example: Typing* logout *ends terminal session.*

Q3: What type of function can be performed in file?

- **Create / Delete files.**
- **Read / Write / Edit contents.**
- **Change permissions.**
  - ☐ *Example: Create →* touch notes.txt, *Delete →* rm notes.txt.

Q4: Draw directory tree of files?

- **Tree Structure:**
- /
- ├── bin
- ├── etc
- ├── home
- │ ├── ali
- │ │   └── file1.txt
- └── usr

☐ Root / at top, branches for sub-directories.

Q5: List down the file types in UNIX?

- **Regular file (–) →** text, binary, program files.
- **Directory (d) →** stores other files.
- **Special files:**
  - o Character devices (c) → keyboard.
  - o Block devices (b) → hard disk.
- **Symbolic link (l) →** shortcut.

Q6: List down characters used as part of the UNIX filenames?

- Allowed: **a–z, A–Z, 0–9, dot (.), underscore (_), dash (-)**.
- Not Allowed: Special chars like / (used for directory), ?, *, |.

Q7: Write command to create directory?

- mkdir directory_name
  - ☐ *Example:* mkdir myfolder

Q8: Write command to remove directory?

- rmdir directory_name → removes empty directory.
- rm -r directory_name → removes directory with files.
  - ☐ *Example:* rmdir oldfiles

Q9: How many types of permissions are there? Write them down?

- **Three types of permissions:**
  1. **Read (r):** View contents.
  2. **Write (w):** Modify contents.
  3. **Execute (x):** Run program or access directory.
     - □ *Example:* `-rwxr--r--` → *Owner (read/write/execute), Group (read), Others (read).*

Q10: What is the purpose of directory name `/bin`?

- Contains **essential binary programs** (commands).
- Accessible to all users.
  - □ *Example:* `ls, cp, mv, rm` *are stored in* `/bin`.

▣ Long Questions with Answers

Q1: How to configure environment in UNIX?

- **Environment Variables:** Store user preferences.
- `PATH` → directories for command search.
- `HOME` → user's default directory.
- `PS1` → shell prompt style.
- **Files used:** `.profile`, `.bashrc` (auto executed at login).
  - □ *Example: Adding* `export PATH=$PATH:/usr/local/bin` *allows using extra commands.*

Q2: Explain File system in UNIX.

- **Hierarchical structure** starting from root `/`.
- **Directories:**
  - o `/bin` → basic commands.
  - o `/etc` → configuration files.
  - o `/home` → user files.
  - o `/usr` → applications.
- **File Types:** regular, directory, device, symbolic link.
- **Advantages:** efficient, secure, organized.
  - □ *Example: File* `/home/ali/test.txt` *is stored under user Ali's directory.*

Q3: Write a detailed note on File Types.

- **Regular Files:**
  - o Store text, program, data.
  - o Example: `notes.txt`.
- **Directories:**
  - o Store other files.
  - o Example: `/home/ali/`.
- **Special Files:**
  - o Character (keyboard), Block (disk).
- **Symbolic Links:**
  - o Shortcuts to other files.
- **Socket & Pipe Files:**
  - o For inter-process communication.

Q4: What do you know about Naming Files and Directories?

- Names can use **letters, numbers, dot, underscore**.
- Cannot use / inside names.
- Case-sensitive: `File.txt` ≠ `file.txt`.
- Max length usually **255 characters**.
  - □ *Example:* `report1.txt, student_data, os-book.pdf` *are valid names.*

Q5: Write a note on File System Organization.

- UNIX uses **tree structure starting with root /.**
- **Root Directory (/)** → base of system.
- Subdirectories like /bin, /etc, /home, /usr.
- **Mounting:** other file systems can be attached.
- Provides **security** through permissions.
  - □ *Example: /home/ali/notes.docx → Root → Home → Ali → File.*

Lect. Malik Omer Asghar