University of Barishal

Department of Computer Science and Engineering

Course Title: System Programming

Course Code: CSE-4105

4th Year 1st Semester (B.Sc.) Final Examination

Admission Session: 2019-2020

Time: 02 Hours

Marks: 60

N.B.: Answer any **FOUR** questions out of the followings. All parts of each question must be answered consecutively. Right side of the question shows the maximum marks.

A user named Asif creates a shell script named backup.sh in their home directory. By default, the file has the permission rw-r--r-- (644). Asif changes the permission using:

```
chmod 754 backup.sh
```

The file permissions are:	**Three users interact with the file:**
- **Owner (Asif):** Read, write, and execute - **Group (Asif's group):** Read and execute - **Others:** Read only	- **Asif**, the owner, wants to run and modify the file. - **Rahim**, a member of the same group as Asif, wants to read and run the file but not change it. - **Karim**, an unrelated user, wants to open, edit, and execute the file.

1.a) Explain in detail how the chmod 754 command changes file permissions. Break down the numeric mode and describe what each number (7, 5, and 4) means in binary, and which permissions they correspond to for the file owner, group, and others. **4**

b) Assuming the default umask is 022, what would the original permissions of **`backup.sh`** be right after it was created? Explain how umask affects file creation permissions and why Asif might have needed to change them **4**

c) If Asif wants to allow only themselves to read, write, and execute the file while preventing everyone else (group and others) from even reading it, what **`chmod`** command should be used? Describe the reasoning and verify the new permission set with symbolic representation (rwxr--r--, etc.). **4**

2. a) A Linux system administrator is investigating a multi-user server to track who is currently logged in and how to monitor system access. The who command is used to list currently logged-in users by reading the **/var/run/utmp** file, which stores login session information. **12**

To build a custom tool that mimics the who command, a developer decides to read /var/run/utmp directly using **low-level Unix system calls**: open(), read(), and close(). Here's how the system calls might be used:

System Call	Purpose
`open()`	Opens the file descriptor to `utmp`
`read()`	Reads binary data from the file
`close()`	Closes the file descriptor

Describe how the who command works internally on Unix-based systems, and explain the role of the low-level system calls open(), read(), and close() in this process.

Your answer should include:

- What file the who command reads (/var/run/utmp) and what kind of information it contains.
- The use of struct utmp to interpret the binary data stored in the file.
- A comparison between low-level system calls (open(), read(), close()).

3.a) A system administrator creates a new user account named alice on a Unix-based system. This user creates several files in her home directory: 12

```
touch /home/alice/report.txt
```

Later, alice is deleted from the system. However, her files remain. The output of ls -l /home/alice/ still shows alice as the file owner.

Some days later, a new employee joins, and the administrator creates a new user also named alice, assigning her a new home directory.

Despite having the same username, this new alice is technically a different user from the system's point of view.

Now, answer the following questions:

i) Describe how the ls command retrieves and displays file ownership and permissions.

ii) Explain what happens when a user is deleted and later a new user is created with the same username.

iii) Given the situation where the original alice had UID 1001 and the new alice is assigned UID 1002, describe what will happen to the files created by the old user.

4.a) When a new file is created on a Unix-based system, the kernel must allocate both an inode and one or more data blocks. Explain how the file system determines which inodes and disk blocks are available for allocation. In your explanation, discuss the data structures or mechanisms used to track free inodes and blocks (such as bitmaps or free lists), and describe how modern file systems (e.g., ext4) manage this process efficiently. 8

b) **cd** *[<**dir**>] means that change* **pwd** to HOME directory, or <**dir**> if it is supplied. However, *"Changing directories"* and "being in a directory" are imprecise phrases. When you **cd** to a *directory named* **dir**, you may think of yourself as being "in **dir**", but this is not true. What is *the true meaning for* **cd** [<**dir**>] in order to resolve the relative pathnames? 4

5.a) *Describe how thread creation and management is handled in POSIX-compliant systems using the pthread library.* 4

b) What are the potential problems of using multiple threads in a shared-memory environment, and how can these problems be mitigated? 4

c) Explain the difference between a process and a thread in the context of system programming. 4

6.a) To a programmer, a system call looks like any other call to a library procedure. Is it important that a programmer know which library procedures result in system calls? Under what circumstances and why? 4

b) A very significant issue for system programming is the handle of I/O devices for accessing the abstraction. Differentiate the access criteria of I/O devices for UNIX and C. 4

c) Depict the control flow of system calls. Do you agree/disagree that the system calls are time consuming? Give your statement. How do you change file permissions in Unix Explain with example. 4