NORTH SOUTH UNIVERSITY

Department of Electrical and Computer Engineering

B.Sc. in Computer Science and Engineering Program

Project (Group), Summer 2023 Semester

Course:	**CSE 323 Operating Systems Design, Section 11**
Instructor:	**Dr. Md. Hasanul Ferdaus, Assistant Professor (Adjunct), ECE Department**
Full Marks:	**25 (25% weight of the total marks)**
Deadline:	**19th November 2023 at 11:59 PM**

Project Title: File Manipulation using System Calls in Ubuntu Linux Operating System

Introduction

The project aims to provide students with a hands-on experience working with system calls on Ubuntu Linux to perform basic file manipulation operations. By interacting directly with the operating system kernel through system calls, students will gain a fundamental understanding of how these low-level interactions work.

Project Features

- **File Creation**: Utilize the ***open*** system call to create a new file.
- **File Writing:** Use the ***write*** system call to write content to a file.
- **File Reading:** Implement the ***read*** system call to display the content of a file.
- **File Appending:** Utilize the ***write*** system call with the ***O_APPEND*** flag to append content to an existing file.
- **File Deletion:** Implement the ***unlink*** system call to delete a file.

Project Design

Data Structures

- **filename:** A character array to store the name of the file.
- **content:** A character array to store the content to be written/read from the file.
- **fileDescriptor:** An integer to store the file descriptor returned by the open system call.

User Interaction

- **File Creation and Writing:** Prompt the user for the file name and content to be written.
- **File Reading:** Display the content read from the file.
- **File Appending:** Prompt the user for additional content to be appended to the file.
- **File Deletion:** Delete the file created earlier.

Implementation

File Creation and Writing:

- Use ***open()*** with appropriate flags and permissions to create a new file.
- Use ***write()*** to write the content to the file.
- Close the file using ***close()*** after writing.

File Reading

- Use ***open()*** with appropriate flags to open the existing file.
- Use ***read()*** to read the content from the file.
- Display the content read from the file to the user.

File Appending

- Use ***open()*** with ***O_APPEND*** flag to open the file in append mode.
- Use ***write()*** to append additional content to the file.
- Close the file after appending.

File Deletion

- Use ***unlink()*** to delete the file.

Deliverables

- **Source Code:** A well-commented C code implementing the specified file manipulation operations using system calls. The filename must follow the format: **GroupXX.c** where XX will be replaced by the group number (For example, for group 4 the filename must be Group04.c and for group 12 the filename must be Group12.c).
- **Project Report:** A detailed report file must be submitted as a MSWord file (DOCX). The filename must follow the format: **ReportXX.docx** (where XX will be replaced by the

group number (For example, for group 4 the filename must be Report04.docx and for group 12 the filename must be Report12.docx).

The report must contain the following sections:

1. A cover page containing the project title, course details (course code and course name), and group details (group number, student names, and student IDs).
2. A detailed documentation explaining the code logic, system calls used, and their purposes.
3. Instructions on how to compile and run the code, along with sample usage scenarios.
4. A summary report on the implementation process, challenges faced, and lessons learned.

Submission Guidelines

- Submit the C file and the report file using the Canvas project submission link by the deadline.

Marking Guidelines

- **Functionality (40%):** The correct implementation of file creation, reading, writing, appending, and deletion using appropriate system calls.
- **Code Quality and Comments (30%):** The clarity of code, proper use of comments, and adherence to coding standards.
- **Error Handling (15%):** Handling of errors such as file not found, permission issues, and unsuccessful system calls.
- **Project Report (15%):** The quality of documentation explaining the code logic, system calls used, and their purposes; instructions on how to compile and run the code, and the sample usage scenarios; and the quality of the summary report on the implementation process, challenges faced, and lessons learned.