National University of Computer and Emerging Sciences, Lahore Campus

Course Name:	Operating Systems - LAB	Course Code:	CL2006
Program:	BS-4G	Semester:	Spring 2024
Duration:	1 Hour 30 Mins	Total Marks:	60
Paper Date:		Weight:	
Section:		Pages:	2
Exam:	Midterm		

Student : Name:______________________________ Roll No.________________ Section:_______

Instruction/Notes:

1. Understanding the question paper is also part of the exam, so do not ask for any clarification.
2. Talking/Discussion is not allowed. You are responsible for protecting your code and saving it from being copied. If you don't protect it all matching codes are considered copy/cheating cases.
3. Failure to observe above mentioned instructions will lead to a negative mark on the Exam.

Question 1: File Manipulation and Inter-Process Communication Challenge [Marks 30]

You are tasked with creating a set of programs for file manipulation and inter-process communication. Your task involves creating four programs: **file_reader**, **word_reverser**, **file_writer**, and **main_program**. Additionally, you need to create a Makefile to build these programs.

1. **file_reader:**
   - This program reads data from an ordinary pipe and writes it to an output file.
2. **word_reverser:**
   - This program reads data from an input file, reverses the order of the words in each line, and writes the result to an output file.
3. **file_writer:**
   - This program reads data from an input file and writes it to a pipe.
4. **main_program:**
   - This program performs the following tasks:
     - Accepts two command-line arguments: an input file name and an output file name.
     - Creates a pipe for IPC.
     - Forks two child processes:
       - In the first child process:
         - Executes the **file_reader** program with the pipe as input and a temporary output file as an argument.
       - In the second child process:
         - Executes the **file_writer** program with the input file as an argument and the pipe as output.
     - In the main process (parent process):
       - Waits for both child processes to complete.
       - Executes the **word_reverser** program with the temporary output file as input and the final output file as output.
       - Removes the temporary output file.
     - Handles errors, such as incorrect number of arguments, file opening issues, fork failures, or pipe creation issues.

Example:

Input file (**input.txt**):

```
Hello, World!
This is an OS section.
```

Output file (**output.txt**):

```
World! Hello,
section. OS an is This
```

Requirements:

- Ensure each program performs its designated task correctly.
- Implement error handling for various scenarios.
- Provide a Makefile to build the programs.

Question 2: File Data Transformation with Doubling, Sorting, and Uniqueness Enforcement [Marks 30]

You are tasked with implementing a C/C++ program that performs the following operations on data stored in a file:

1. Double the values in the first column of the file.
2. Sort the data in human-readable numeric order.
3. Ensure that only unique entries are retained.

The terminal command that accomplishes this task is:

```
awk '{print $1*2}' input_file.txt | sort --human-numeric-sort | uniq > output_file.txt
```

Your program should take two parameters: the input file name (**input_file.txt**) and the output file name (**output_file.txt**). Implement the necessary logic in C/C++ to execute the operations described above on the data from the input file and write the results to the output file. Ensure that the program handles any errors gracefully and provides appropriate feedback to the user in case of failures.