**Week 4:**

Build a file-processing application in C++ using multithreading.

* Multithreaded program to process large text files concurrently.
* Threads should perform following tasks: counting words, counting punctuations, identifying & counting date patterns.
* File I/O for reading input and writing results to an output file.
* Exception handling for file errors or invalid input.
* Execution screenshot along with logs.

Code Repository: <https://github.com/Akriti987/Weekly_Tasks.git>

**Execution Documentation for C++ Hotel-Booking System**

This document describes a C++ application implementing a file processing System using file handling operations.

This project aims to create a simple command-line interface for users to:

* Take input from ‘input.txt’ file and process it.
* Count words.
* Count punctuations.
* Count dates.
* View output from ‘output.txt’ file.

**1. Build Instructions**

1. Save the code as **file\_processor.cpp**
2. Compile the code using a C++ compiler

**2. Running the Application:**

* **Compile the code using a C++ compiler:**

g++ -std=c++11 file\_processor.cpp -o file\_processor -pthread.

* **Run the executable:**

./file\_processor

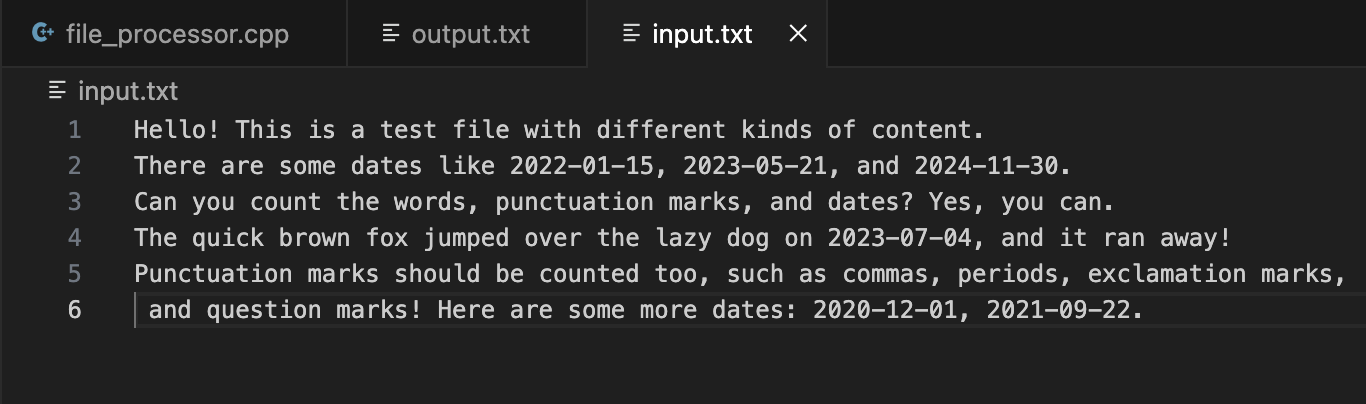
**4.Execution Screenshots**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.**

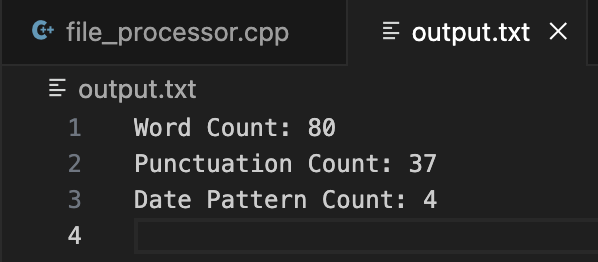
****

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.**

****

**5.Code Snippet**

A screen shot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.A screen shot of a computer program

AI-generated content may be incorrect.