Challenge: Log Analyzer and Report Generator

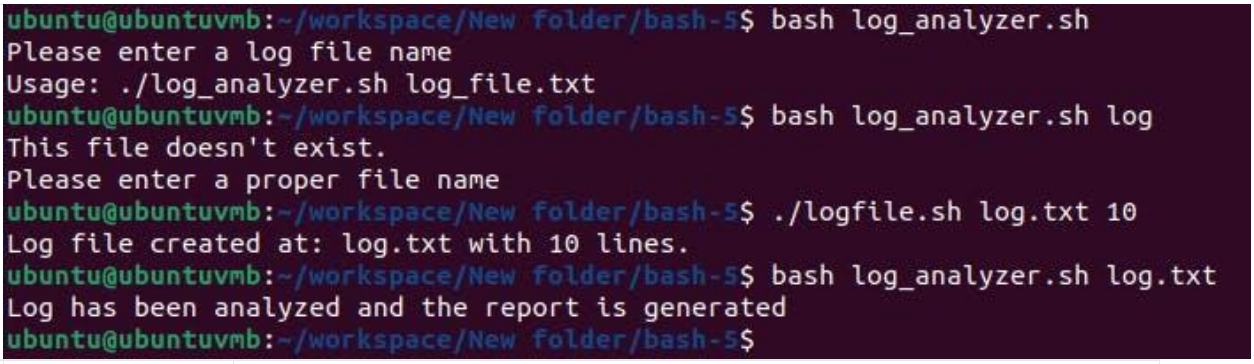
Scenario

You are a system administrator responsible for managing a network of servers. Every day, a log file is generated on each server containing important system events and error messages. As part of your daily tasks, you need to analyze these log files, identify specific events, and generate a summary report.

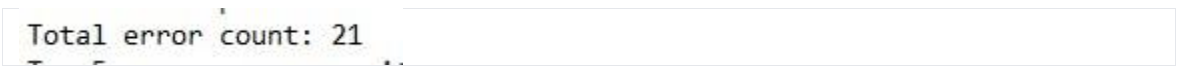
Task

Write a Bash script that automates the process of analyzing log files and generating a daily summary report. The script should perform the following steps:

1. **Input:** The script should take the path to the log file as a command-line argument.



2. **Error Count:** Analyze the log file and count the number of error messages. An error message can be identified by a specific keyword (e.g., "ERROR" or "Failed"). Print the total error count.



3. **Critical Events:** Search for lines containing the keyword "CRITICAL" and print those lines along with the line number.

```
List of critical events with line numbers:
Line No. 7 : 2023-08-04 21:01:50 [CRITICAL]  - 1766
Line No. 11 : 2023-08-04 21:01:50 [CRITICAL]  - 11674
Line No. 13 : 2023-08-04 21:01:50 [CRITICAL]  - 23117
Line No. 16 : 2023-08-04 21:01:50 [CRITICAL]  - 15256
Line No. 24 : 2023-08-04 21:01:50 [CRITICAL]  - 27423
Line No. 27 : 2023-08-04 21:01:50 [CRITICAL]  - 21693
Line No. 36 : 2023-08-04 21:01:50 [CRITICAL]  - 7130
Line No. 40 : 2023-08-04 21:01:50 [CRITICAL]  - 3756
Line No. 48 : 2023-08-04 21:01:50 [CRITICAL]  - 11533
Line No. 56 : 2023-08-04 21:01:50 [CRITICAL]  - 4344
Line No. 64 : 2023-08-04 21:01:50 [CRITICAL]  - 9403
Line No. 72 : 2023-08-04 21:01:50 [CRITICAL]  - 4871
Line No. 76 : 2023-08-04 21:01:50 [CRITICAL]  - 28076
Line No. 84 : 2023-08-04 21:01:50 [CRITICAL]  - 26040
Line No. 88 : 2023-08-04 21:01:50 [CRITICAL]  - 28114
```

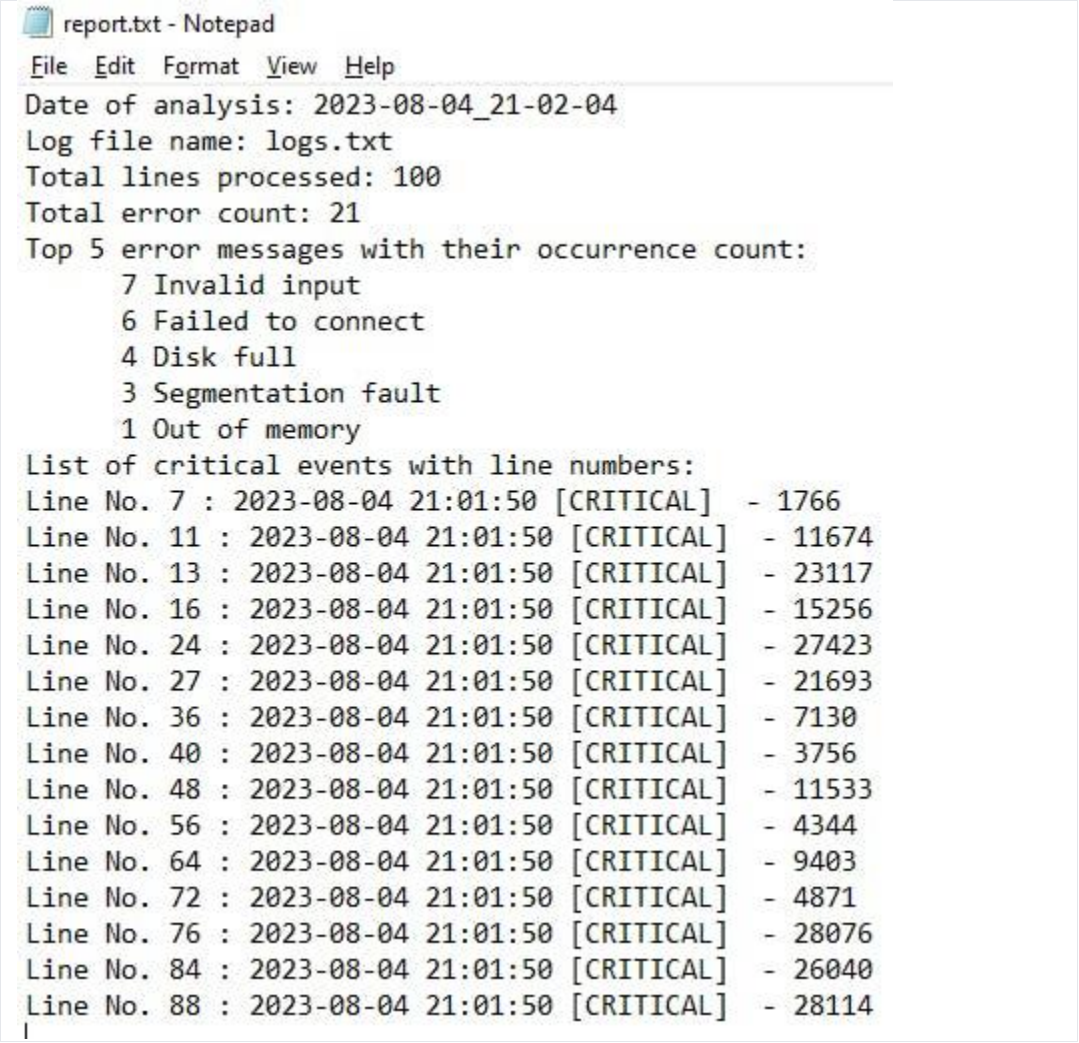
4. **Top Error Messages:** Identify the top 5 most common error messages and display them along with their occurrence count.

```
Top 5 error messages with their occurrence count:
      7 Invalid input
      6 Failed to connect
      4 Disk full
      3 Segmentation fault
      1 Out of memory
```

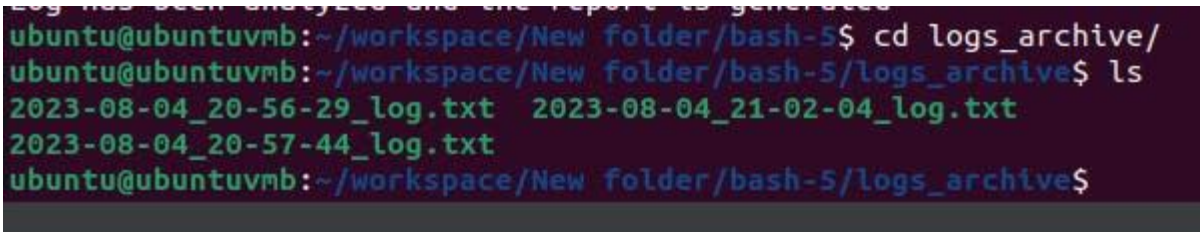
5. **Summary Report:** Generate a summary report in a separate text file. The report should include:

   - Date of analysis
   - Log file name
   - Total lines processed
   - Total error count
   - Top 5 error messages with their occurrence count

- List of critical events with line numbers



Optional Enhancement: Add a feature to automatically archive or move processed log files to a designated directory after analysis.



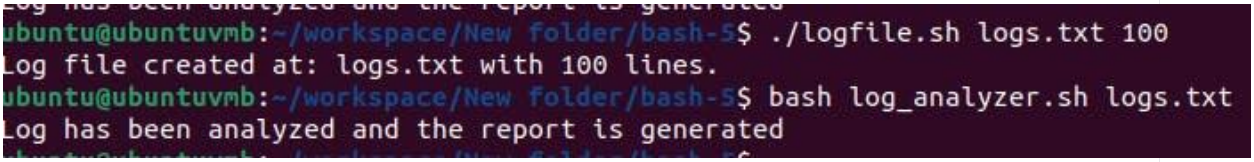
Usage of the script:

1. Create a log file using the logfile.sh-

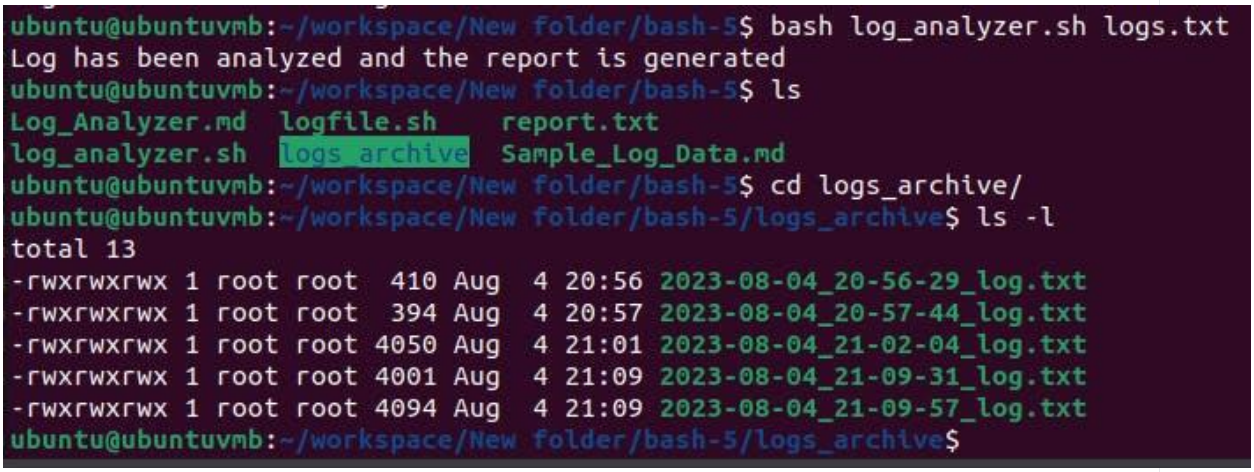
./logfile.sh <log_filename> <no_of_logs_in_file>

2. Use the log_anayzer.sh to create the report.

./log_analyzer.sh <log_filename>



You will find the report in report.txt and the log files will be renamed according to the date and time of analysis in the logs_archive folder.



The code -

```bash
#!/bin/bash
if [ ${#1} -eq 0 ]
then
    echo "Please enter a log file name"
    echo "Usage: ./log_analyzer.sh log_file.txt"
else
    file=$1
    if [ -f "$file" ];
    then
        i=1
        no_of_error=0
        date_of_analysis=$(date '+%Y-%m-%d_%H-%M-%S')
        lines=`cat $file | wc -l`
        no_of_error=`cat $file | grep -E "ERROR|FAILED" | wc -l`
        top_error=`cat $file | awk -F'] |- ' '{print $2}' | sort | uniq -c | sort -nr | head -n 6 | tail -n 5`

        echo "Date of analysis: $date_of_analysis" > report.txt
        echo "Log file name: $file" >> report.txt
        echo "Total lines processed: $lines" >> report.txt
        echo "Total error count: $no_of_error" >> report.txt
        echo "Top 5 error messages with their occurrence count: " >> report.txt
        echo "$top_error" >> report.txt

        while read line; do
            if [[ $line == *"CRITICAL"* ]]; then
                echo "Line No. $i : $line" >> report.txt
            fi
            i=$((i+1))
        done < $file
        dest_path="logs_archive"
        mv $file "$date_of_analysis""_log.txt"
        file="$date_of_analysis""_log.txt"
        mkdir -p $dest_path
        mv $file $dest_path
        echo "Log has been analyzed and the report is generated"
    else
    echo "This file doesn't exist."
    echo "Please enter a proper file name"
    fi
fi
```

Thanks!