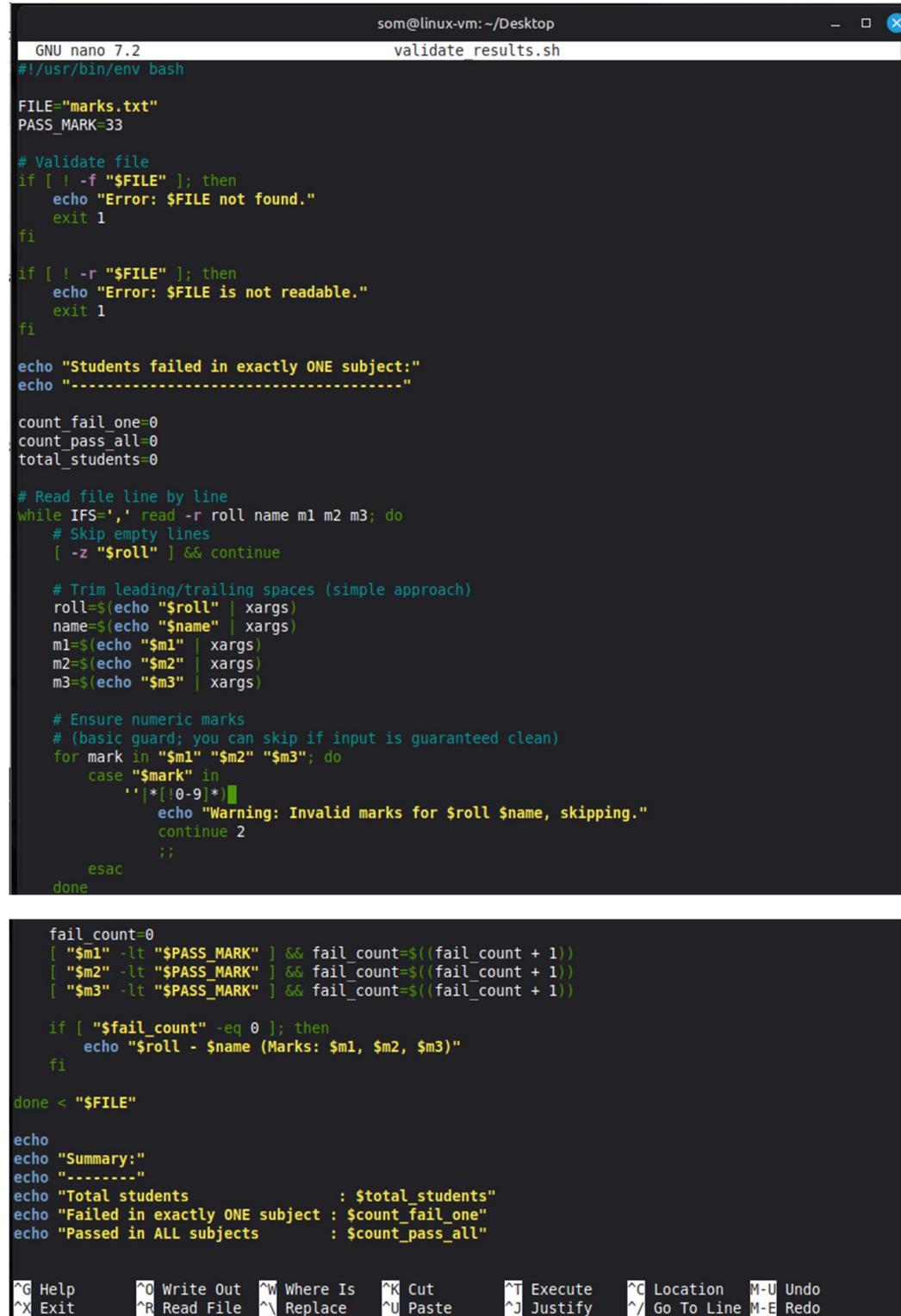


### Question 3 (2024eb03003):

Please find screenshot of shell script below and attaching **validate\_results.sh** script to GitHub repository:



The screenshot shows a terminal window titled "validate results.sh" running on a Linux VM. The window contains the source code of a shell script named "validate\_results.sh". The script uses the GNU nano 7.2 editor. The code is a script to validate student marks from a file named "marks.txt". It checks if the file exists and is readable. It then reads the file line by line, trimming spaces and ensuring numeric marks. It counts students failing exactly one subject and those failing all subjects. Finally, it prints a summary with the total number of students, the count of students failing exactly one subject, and the count of students passing all subjects.

```
GNU nano 7.2 validate results.sh
#!/usr/bin/env bash

FILE="marks.txt"
PASS_MARK=33

# Validate file
if [ ! -f "$FILE" ]; then
    echo "Error: $FILE not found."
    exit 1
fi

if [ ! -r "$FILE" ]; then
    echo "Error: $FILE is not readable."
    exit 1
fi

echo "Students failed in exactly ONE subject:"
echo "-----"

count_fail_one=0
count_pass_all=0
total_students=0

# Read file line by line
while IFS=',' read -r roll name m1 m2 m3; do
    # Skip empty lines
    [ -z "$roll" ] && continue

    # Trim leading/trailing spaces (simple approach)
    roll=$(echo "$roll" | xargs)
    name=$(echo "$name" | xargs)
    m1=$(echo "$m1" | xargs)
    m2=$(echo "$m2" | xargs)
    m3=$(echo "$m3" | xargs)

    # Ensure numeric marks
    # (basic guard; you can skip if input is guaranteed clean)
    for mark in "$m1" "$m2" "$m3"; do
        case "$mark" in
            ''|*[!0-9]*)#
                echo "Warning: Invalid marks for $roll $name, skipping."
                continue 2
            ;;
        esac
    done

    fail_count=0
    [ "$m1" -lt "$PASS_MARK" ] && fail_count=$((fail_count + 1))
    [ "$m2" -lt "$PASS_MARK" ] && fail_count=$((fail_count + 1))
    [ "$m3" -lt "$PASS_MARK" ] && fail_count=$((fail_count + 1))

    if [ "$fail_count" -eq 0 ]; then
        echo "$roll - $name (Marks: $m1, $m2, $m3)"
    fi

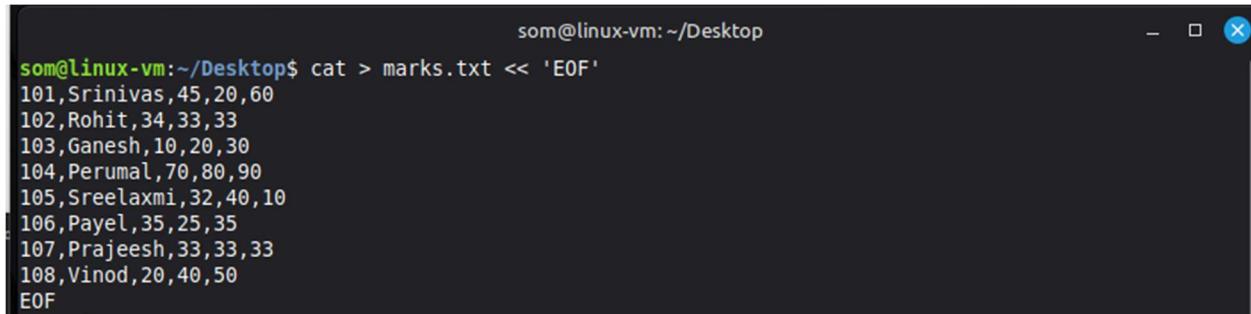
done < "$FILE"

echo
echo "Summary:"
echo "-----"
echo "Total students           : $total_students"
echo "Failed in exactly ONE subject : $count_fail_one"
echo "Passed in ALL subjects      : $count_pass_all"
```

## **Testing the validate\_results.sh Script**

Create a file **marks.txt**

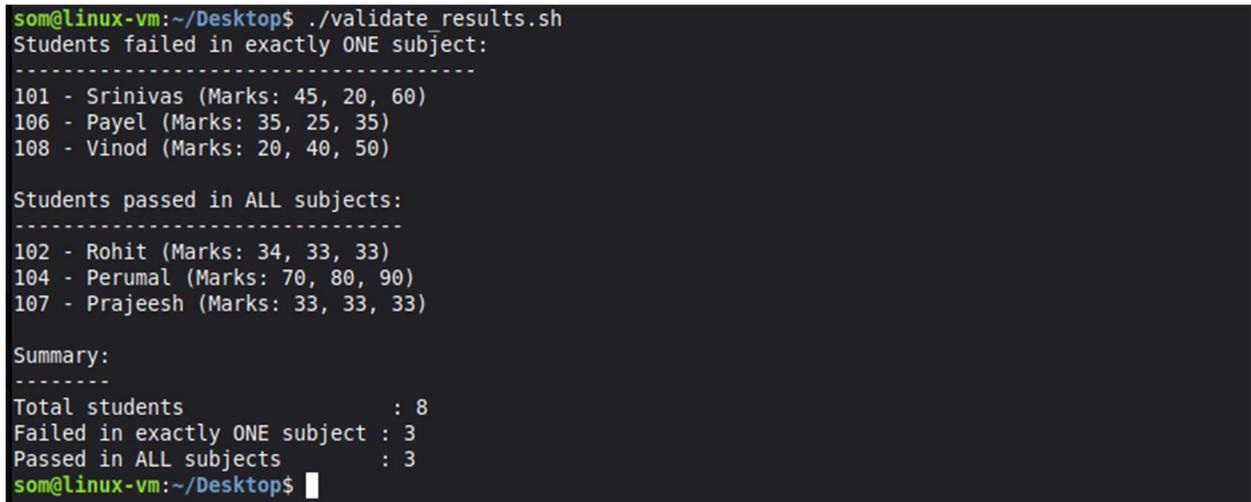
```
cat > marks.txt << 'EOF'  
101,Srinivas,45,20,60  
102,Rohit,34,33,33  
103,Ganesh,10,20,30  
104,Perumal,70,80,90  
105,Sreelaxmi,32,40,10  
106,Payel,35,25,35  
107,Prajeesh,33,33,33  
108,Vinod,20,40,50  
EOF
```



```
som@linux-vm:~/Desktop$ cat > marks.txt << 'EOF'  
101,Srinivas,45,20,60  
102,Rohit,34,33,33  
103,Ganesh,10,20,30  
104,Perumal,70,80,90  
105,Sreelaxmi,32,40,10  
106,Payel,35,25,35  
107,Prajeesh,33,33,33  
108,Vinod,20,40,50  
EOF
```

## **Test Case 1: Normal operation**

**./validate\_results.sh**



```
som@linux-vm:~/Desktop$ ./validate_results.sh  
Students failed in exactly ONE subject:  
-----  
101 - Srinivas (Marks: 45, 20, 60)  
106 - Payel (Marks: 35, 25, 35)  
108 - Vinod (Marks: 20, 40, 50)  
  
Students passed in ALL subjects:  
-----  
102 - Rohit (Marks: 34, 33, 33)  
104 - Perumal (Marks: 70, 80, 90)  
107 - Prajeesh (Marks: 33, 33, 33)  
  
Summary:  
-----  
Total students : 8  
Failed in exactly ONE subject : 3  
Passed in ALL subjects : 5  
som@linux-vm:~/Desktop$
```

**Requirement validated:**

Expected results from this data:

- Failed exactly ONE subject: (03)

101,Srinivas,45,20,60

106,Payel,35,25,35

108,Vinod,20,40,50

- Passed ALL subjects: (03)

102,Rohit,34,33,33

104,Perumal,70,80,90

107,Prajeesh,33,33,33

- Total students: (08)