CS108-Project

Bash Grader

Aakash Gupta
Indian Institute of Technology Bombay

April 26, 2024

Contents

1 Abstract

The bash script **submission.sh** implement various command line functions to handle a large number of csv files and allows the user to implement many other customised tasks despicted in other python files easily.It is basically a ***CSV file manager and interpreter.***

2 Basic Idea to the code

Submitted file **submission.sh** uses various scripting languages like unix command line, sed and awk to implement various tasks explained in following section.Python files attached are for customisations and use various libraries like numpy, pandas, matplotlib, Levenshtein etc to help the user analyse and interpret the data given in csv files.

3 Usage of basic code

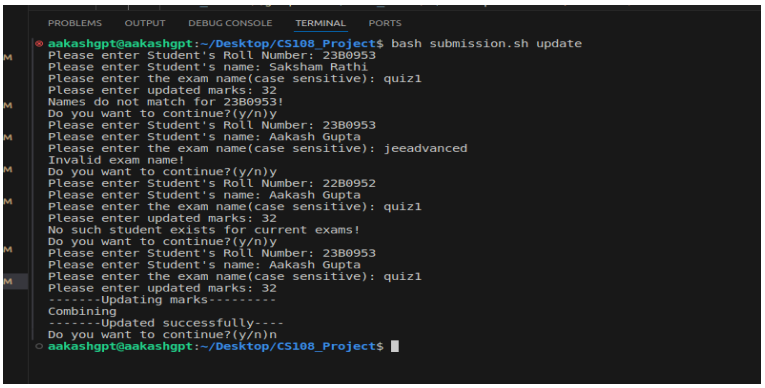
The usage of the files is simple and easy. Open a terminal on your computer. Change the directory to the location where the files are present. Now run the following command on the terminal:

- **bash submission.sh combine**
  This command implement **combine** command which after iterating through all csv files in that directory makes main.csv of format $< Roll\_Number, Name, marks >$ ensuring that students who are absent in particular exam will be assigned 'a'.

- **bash submission.sh upload <Address to file>**
  This provide the provision to **upload** new csv files to directory where submission.sh is present so that after running combine again, main.csv will be updated accordingly.

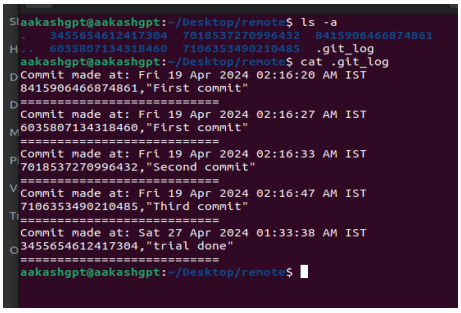
- **bash submission.sh total**
  After running, the awk file will append new column **total** with totalled values of all so that it is still capable of running combine with updating main.csv. Also, after running the command again, the user will be notified that totalling has already been done!

- **bash submission.sh update**
  The yes/no interface allows the user to **update** more number of time. The usage has been depicted in the figure below. The implementation is such that input roll number and name are case insensitive but not so for exam name as there can be more exams with same name but different cases.

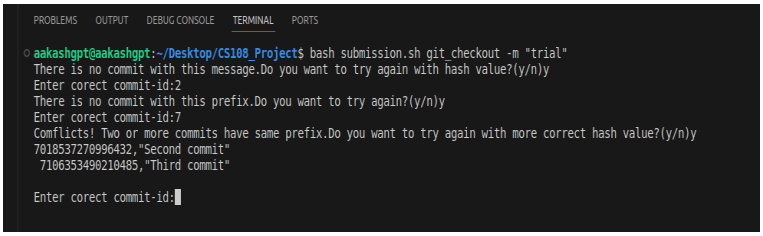


- **bash submission.sh git_init <Address to folder>**
  This command initializes a new **git** directory at given address and if already present, notifies the user about it.

- **bash submission.sh git_commit -m "message"**
  This command copies current version of all files in current directory to remote directory with a particular 16 digit random hash value and the **commited** message and also appends the same into a .git_log file in the remote directory. It also echo all files which have been modified wrt latest commit.



- **bash submission.sh git_checkout -m "commit-message" or "hash-value"**
  This command reverts our current directory back to the commit message we specify. Even if prefix of hash value is given, It will suggest as shown in the image below.



4 Customisations Implemented and Usage

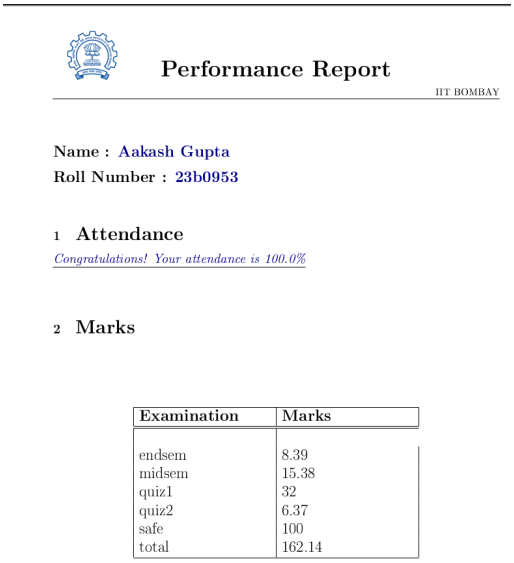
The python files and latex file along with submission.sh contains script for implementing the following customisations:

- **bash submission.sh analysis**
  This command print **analysis** of what the user ask for. Example if input is 1 then print for all exams else if input is 2 then ask for a particular exam. Statistics includes *mean, median, mode, standard deviation, maximum, minimum, 25th percentile and 75th percentile.*

- **bash submission.sh grading**
  Not only making grades.txt having **grades** of all students in main.csv but also implementing relative grading using gaussian distribution and a bell shaped curve, It serves as a powerfull customised tool for analysis student data on large scale.

- **bash submission.sh search**
  Given student's name or roll number(user interface), it will print the whole data of that student from main.csv.

- **bash submission.sh report**
  After getting user input of name or roll number, this will automatically generate pdf file of **personalised performance report** of that student which not only take into consideration sleek and professional look but also present all necessary information and comments about attendance



Performance Report

IIT BOMBAY

Name : Aakash Gupta
Roll Number : 23b0953

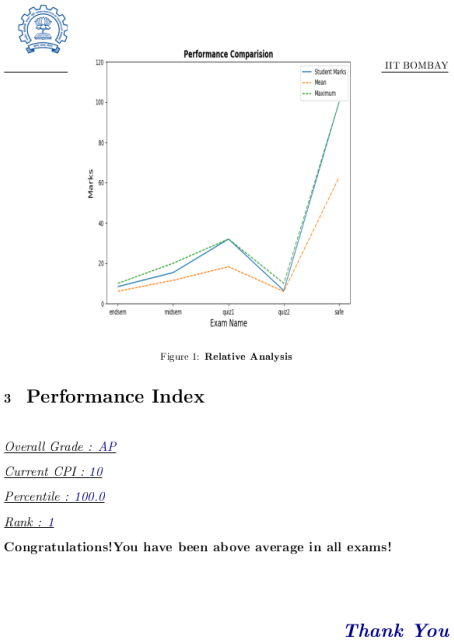
1 **Attendance**

Congratulations! Your attendance is 100.0%

2 **Marks**

Examination	Marks
endsem	8.39
midsem	15.38
quiz1	32
quiz2	6.37
safe	100
total	162.14

(a) Front page



IIT BOMBAY

Figure 1: **Relative Analysis**

3 **Performance Index**

Overall Grade : AP
Current CPI : 10
Percentile : 100.0
Rank : 1

Congratulations!You have been above average in all exams!

Thank You

(b) Back page

- **bash submission.sh general_report**
  This differs from above report in the way that it presents **overall analysis** and all important information about all exams that can help the user in interpreting the overall batch performance.
- **Closest Matching**
  This has been incorporated in search and report command line function. Upon giving wrong student name, It will automatically predict the **closest matching** and help the user in doing the work faster and efficiently. Also the above fucntions ignore case of roll number.
- **progress bar**
  Animated progress bar for combine function for visual appearance.

5 Utilities

Beside **Bash Sed and awk** have also been used. For implementing customisations, **Python, Matplotlib, Numpy and Pandas** refered from [1] have also been used. For making report, **Latex** have also played significant role in completion of this wonderful project.

6 Dependencies

Make sure that before running the following commands, beside scripting languages like bash, sed and awk (referred from Chebrolu [3]), various python modules like pandas, numpy ,matplotlib, os (referred from [2]), sys and Levenshtein are installed on the system.

7 Modularisation

Making separate functions for every command helped in modularising the code. For implementing customisations, making separate files helped in properly utilising the imported functions and variables.

8 LATEX Report

This submitted pdf file has been created using latex with the usage of ***pdflatex report.tex***.

9 Applications

A Bash grader script can be a simple yet effective tool for generating totals, grading, and report cards for students in various educational settings. Here are some applications and use cases for such a tool:

- **Automated Grading:**
  In educational institutions, teachers often need to grade assignments, quizzes, or exams. A Bash grader script can automate this process by parsing student submissions, calculating scores based on predefined criteria, and generating grades.
- **Generating Reports:**
  Once grading is complete, the Bash script can generate individualized report cards or grade reports for each student. These reports can include overall scores, grades, comments, and any other relevant information.
- **Course Management:**
  In addition to grading, a Bash grader script can assist with course management tasks such as maintaining student records, tracking progress, and generating statistical analyses of student performance.
- **Feedback and Improvement:**
  By providing detailed feedback and analysis, the Bash grader script can help students identify areas for improvement and track their progress over time. This can facilitate personalized learning and academic growth.
- **Online Learning Platforms:**
  With the increasing popularity of online learning platforms, a Bash grader script can be integrated into these platforms to automate grading and assessment processes, providing a seamless experience for both students and instructors.
- **Customization and Extensibility:**
  One of the advantages of using a Bash grader script is its flexibility and extensibility. Educators can customize the script to suit their specific grading criteria, assessment methods, and educational objectives.

  Overall, a Bash grader script can streamline the grading process, save time for educators, provide valuable feedback to students, and enhance the overall learning experience.

References

[1] Pandas, Numpy and Matplotlib. URL: https://www.w3schools.com/python/default.asp.

[2] Command line arguments in python. URL: https://www.geeksforgeeks.org/command-line-arguments-in-python/.

[3] Kameswari Chebrolu. *CS108-Course slides-Bash,sed and awk*.