Table of Contents

[Background 1](#_Toc110898414)

[About our Project 1](#_Toc110898415)

[Outcome 2](#_Toc110898416)

[Learning 7](#_Toc110898417)

Table of Figures

[Figure 1 After Running AddEntry.c 2](#_Toc110898559)

[Figure 2 All the error handling and the quit message 3](#_Toc110898560)

[Figure 3 Initial Output from ViewRatings.c 4](#_Toc110898561)

[Figure 4 A student's rating and error handling within it 5](#_Toc110898562)

[Figure 5 Printing All Students Ratings using 717 5](#_Toc110898563)

[Figure 6 Printing AKmaam’s rating from all students 6](#_Toc110898564)

[Figure 7 Printing all teachers ratings and also the average 6](#_Toc110898565)

[Figure 8 Error Handling in Teacher's rating print 7](#_Toc110898566)

[Figure 9 Ending the ViewRatings.c Program 7](#_Toc110898567)

# Background

Since we have had our studies and curriculum run at quite an impressive pace for almost 3 months, we have finished our coursework successfully. Some subjects took a short time while some of them took a little longer. Some classes were so good that even the back benchers would join them. Some were so boring that everyone wanted nothing less than to bunk. But, were we just guessing this as the reality or was there really this perception among our classmates? How can we share our experiences with our juniors? So, It was a good time to reflect upon the teachers and how helpful they have been for us throughout the months. With this impressive thought in our mind, we decided, what would be a better idea than to create a rating software where all of the students can tell us their opinions on the teachers. And yes, that's exactly what we did in this project.

# About our Project

This project was totally made in C language using only one header to showcase the extent and usefulness of simple programming concepts we were actually taught. Nothing beyond the scope of our curriculum was used or implemented in this project. <stdio.h>, a header responsible for standard input and output (file handling) is the only header we have used. Our project uses many, or all the concepts we were taught in out lab and theory classes like variables, printing to user, loops, arrays, pointers, file handling by reading, appending and writing, recursive functions, user defined functions, structures, function call on another function, and many more. This project is just a simple example to display problem solving skill is what matters more than language.

This project of ours consists of 2 parts. One program will help users to add ratings to the teachers. Another will help users view the rating(s) as they need. Though they are separate programs on their own, their many functions and ideas.

The AddEntry.c program is all about adding new entries of ratings. We have scoped the entry to our CD section only to challenge ourselves in adding restrictions in our software. The teachers are also limited to the teachers of our section. However, this can easily be changed by editing simple values within the program. This program manages to provide good error handling. It displays proper error messages to the user if the intended use of the program is not correctly followed. This program needs 2 databases which we will later see also being used in our next program. The databases are Student.txt and rating\_data.txt.

The ViewRatings.c program on the other side helps you view the ratings in several ways users may find useful. Users can view the ratings made by a particular student, view all the ratings made by every student at once, view the ratings received by a particular teacher or view the ratings received by all the teachers at once. This program too handles the errors users might make quite well.

The database rating\_data.txt consists of 2 lines of data where the first line stores the total number of students each teacher has received rating from. The second line of this database stores the sum of rating teachers have received.

The Student.txt database consists of several rows of data where each row holds the rating a student has given to all the teachers. The first two digits show the roll of the student.

Working of both of these software are simple and anyone with basic concepts of programming and C language can easily understand this.

# Outcome

Several screenshots of outputs we could obtain from the code are displayed below.

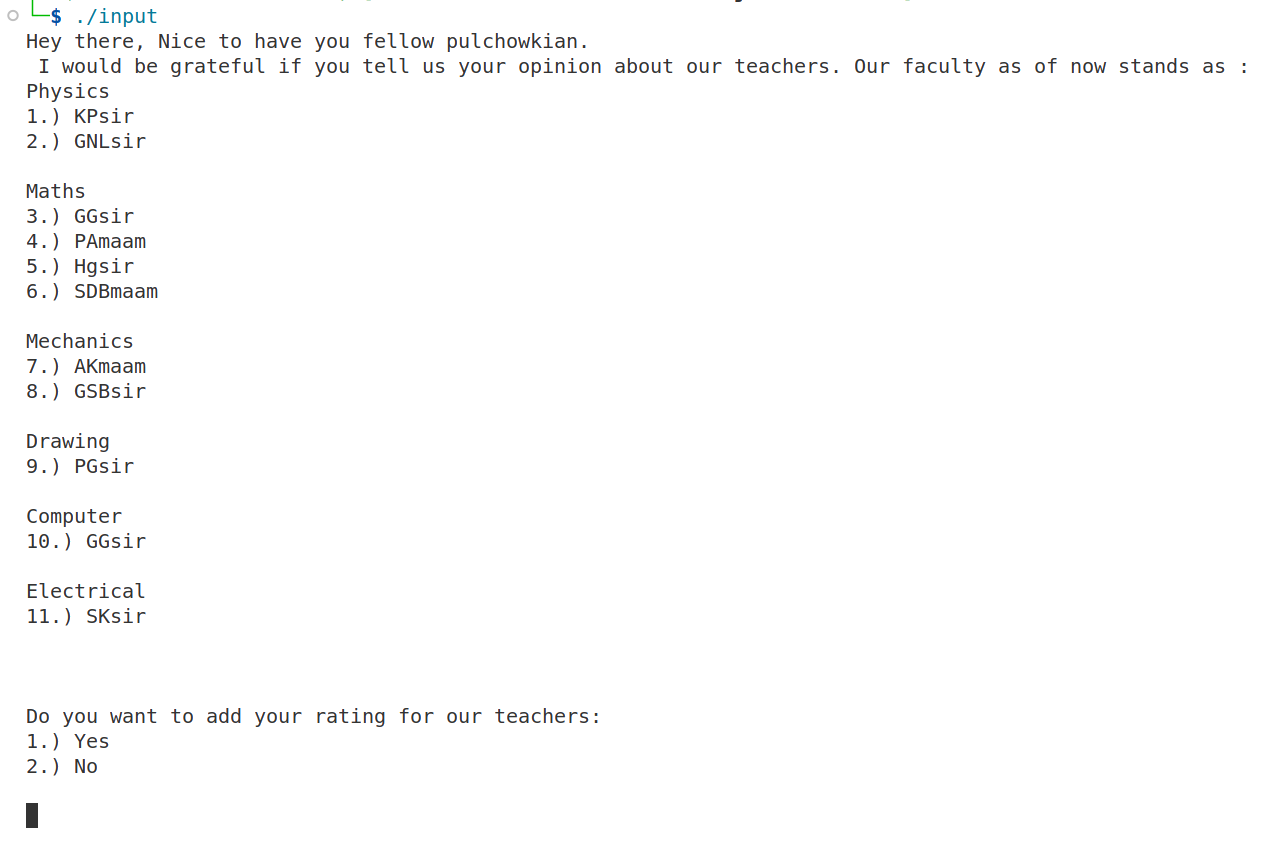


Figure After Running AddEntry.c

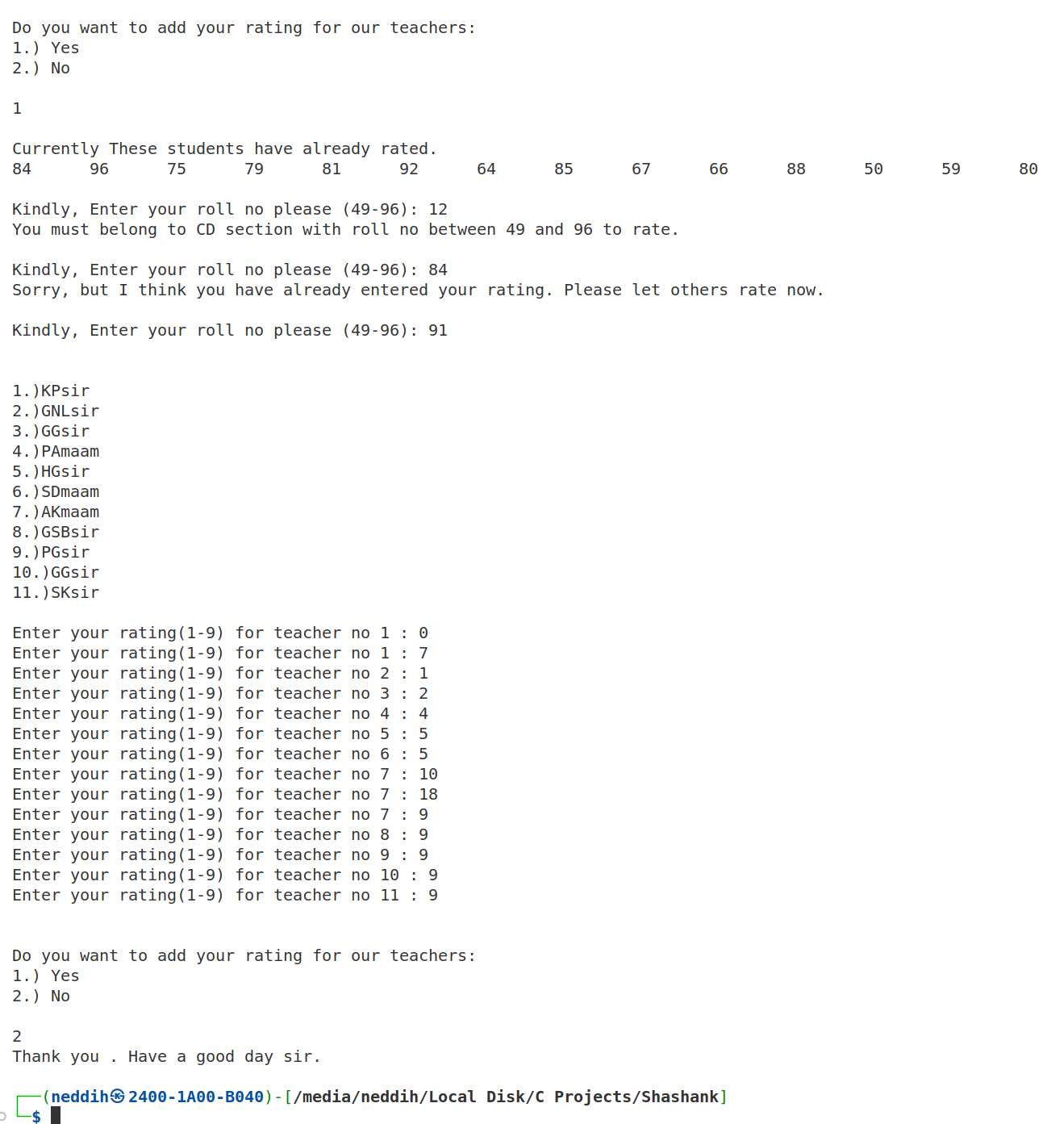


Figure All the error handling and the quit message

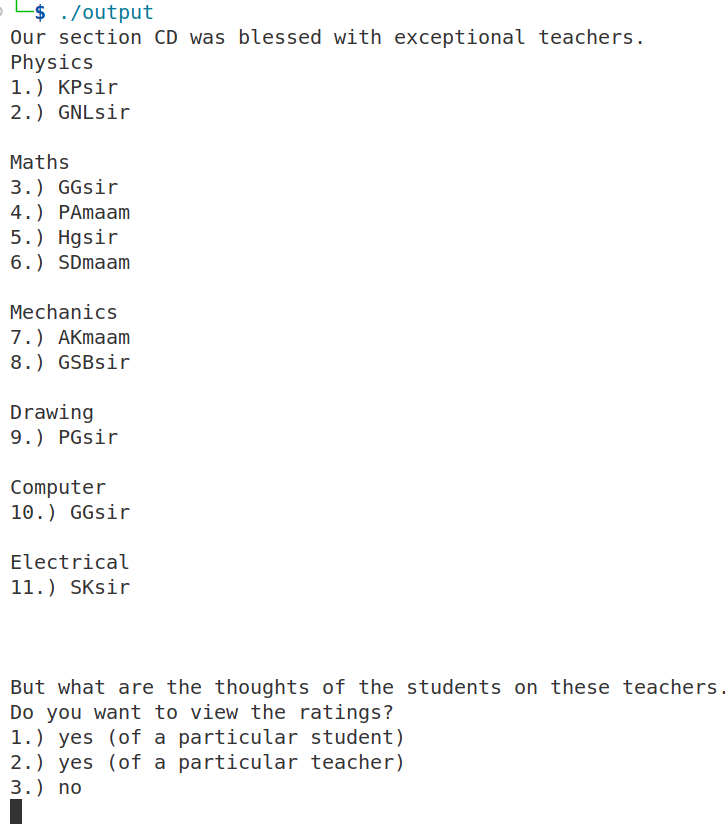


Figure Initial Output from ViewRatings.c

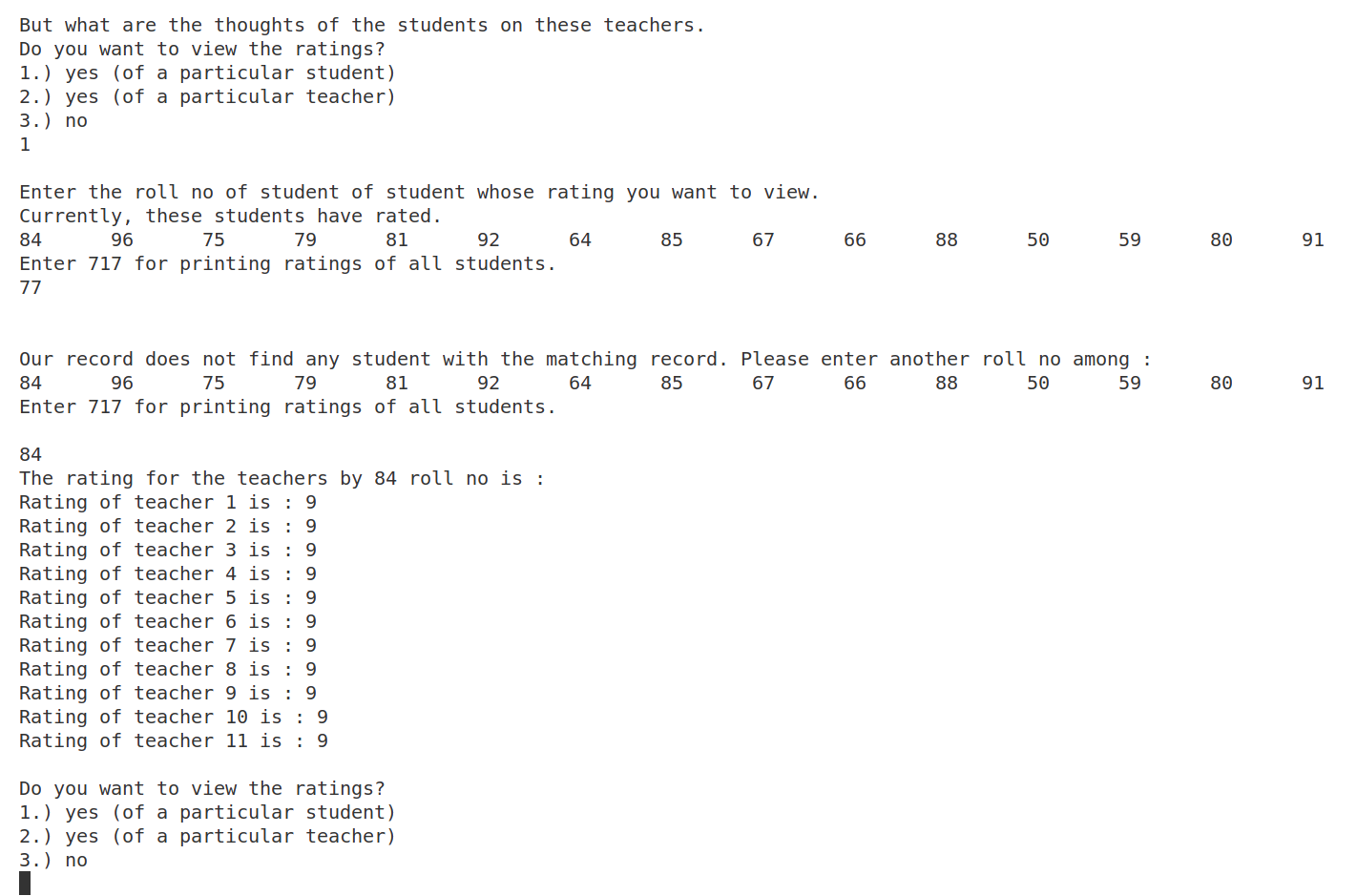


Figure A student's rating and error handling within it

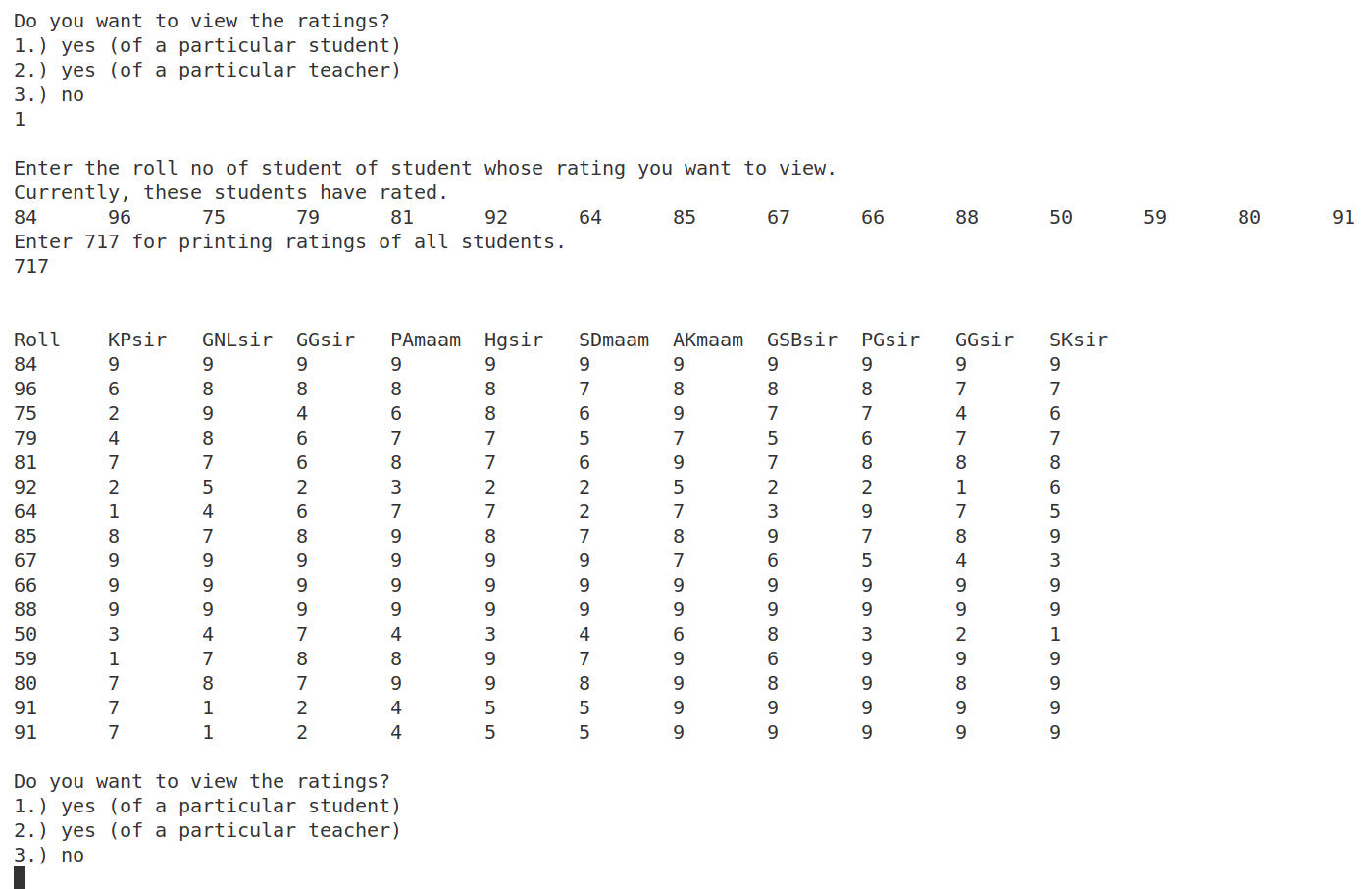


Figure Printing All Students Ratings using 717

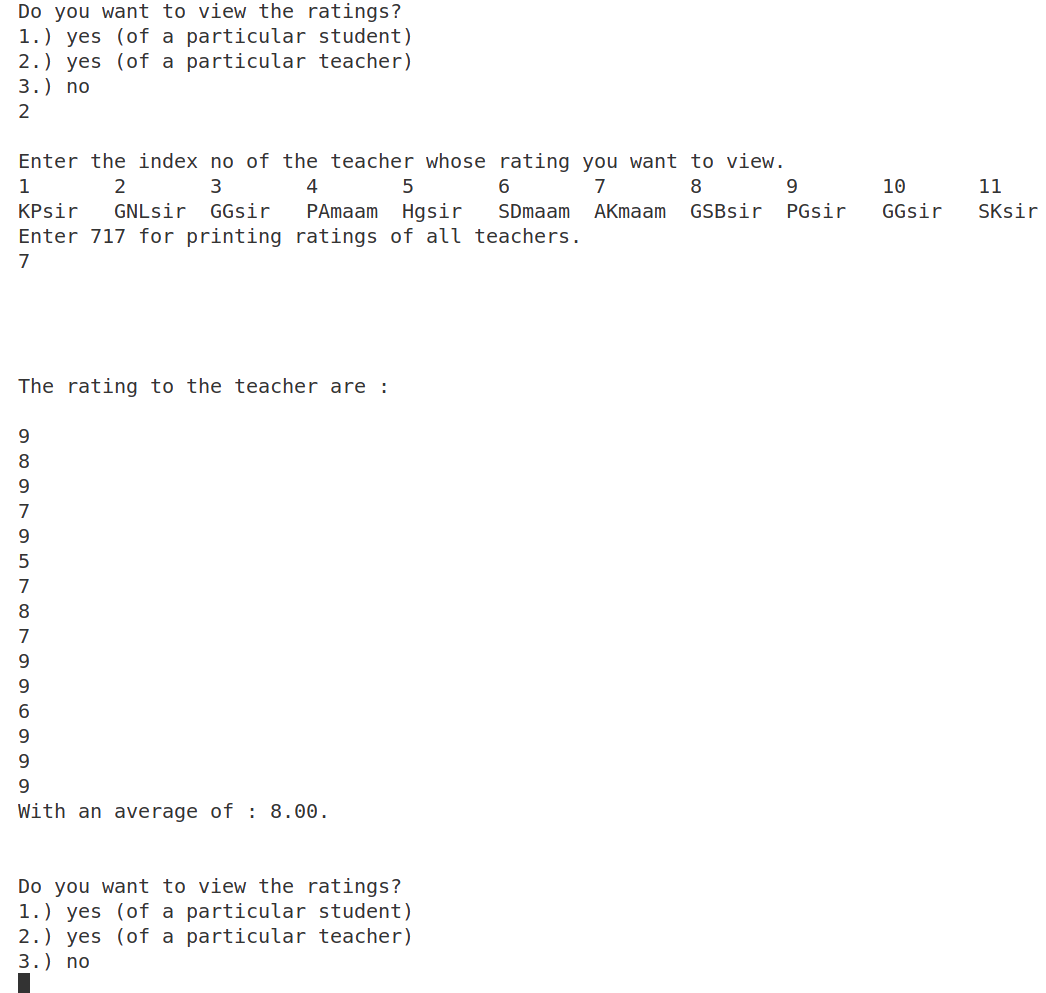


Figure Printing AKmaam’s rating from all students

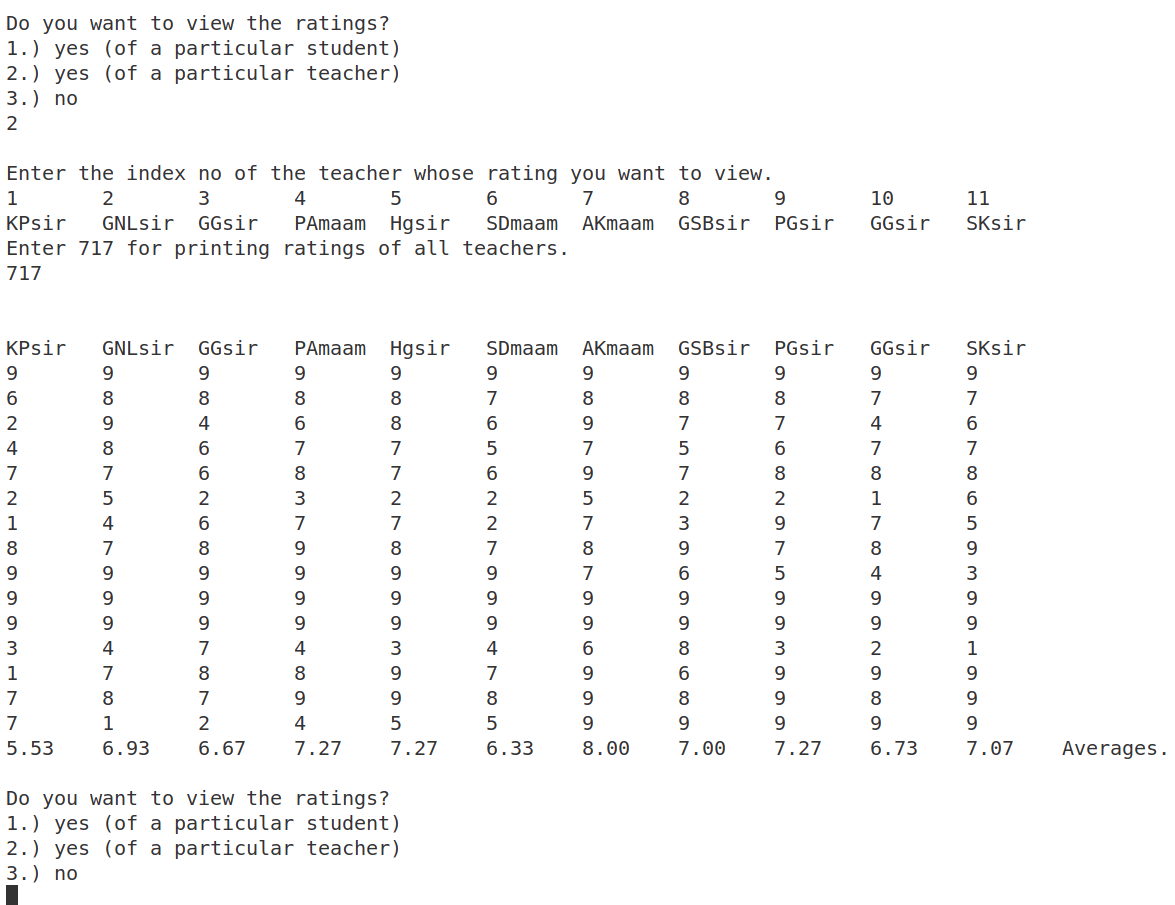


Figure Printing all teachers ratings and also the average

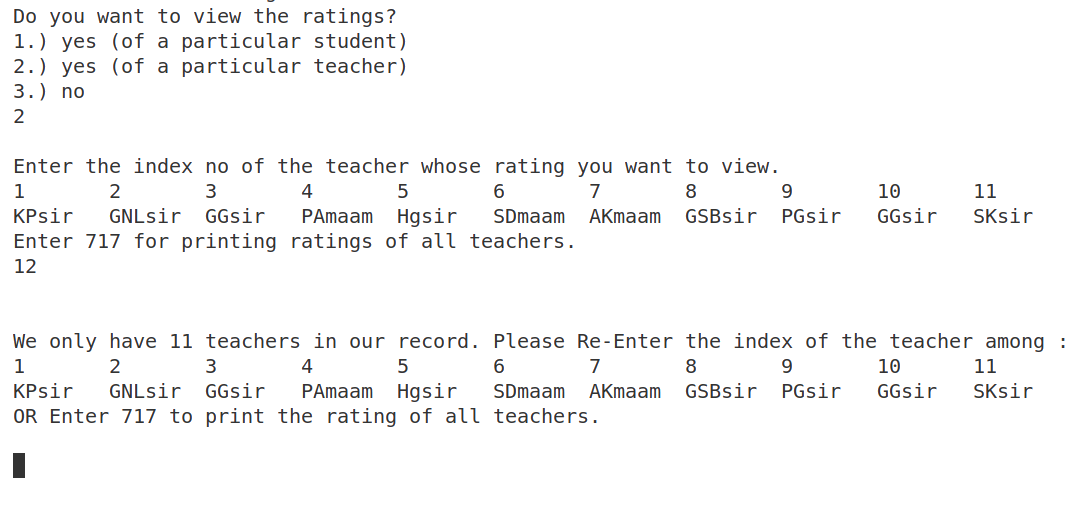


Figure Error Handling in Teacher's rating print

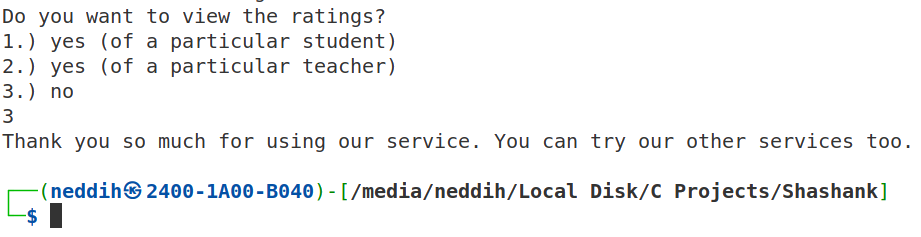


Figure Ending the ViewRatings.c Program

# Learning

By doing this project, We could get a grasp on all the fundamental programming topics and techniques. Utilize and apply all the concepts we learned through these 2 months. It also made us realize that there is still plenty of room to improve upon such as graphical interface, better resource management, etc. It has helped us learn and paved the way for our future in programming and left us with more challenges and excitement for the future.

In general, we learned the use of variables, printing in terminal, loops, arrays, pointers, file handling by reading, appending and writing, recursive functions, user defined functions, structures, function calls and effective use of functions. We also got to learn some practical things like, not everything we think will work actually works. So, we had to debug our way through the semantic errors, correcting our logic and thus helping us to learn various ways of debugging.  We also learned the convenience a proper IDE provided and the importance of proper formatting while coding. We used VS Code with some extensions installed while coding which made our time more efficient.