

OOP Final Project

By John adams

**U08601509 | COP3331-001 | 4/21/18**

# Contents of zip file 3331project.zip

The most important files inside the zip folder are in a folder named “**myprogram**”

Which contains:

* Main.cpp (all source code in this file)
* Main.exe
* passwords.txt and scores.txt
* runproject.bat

**Also inside 331project.zip but outside Final\_project folder**

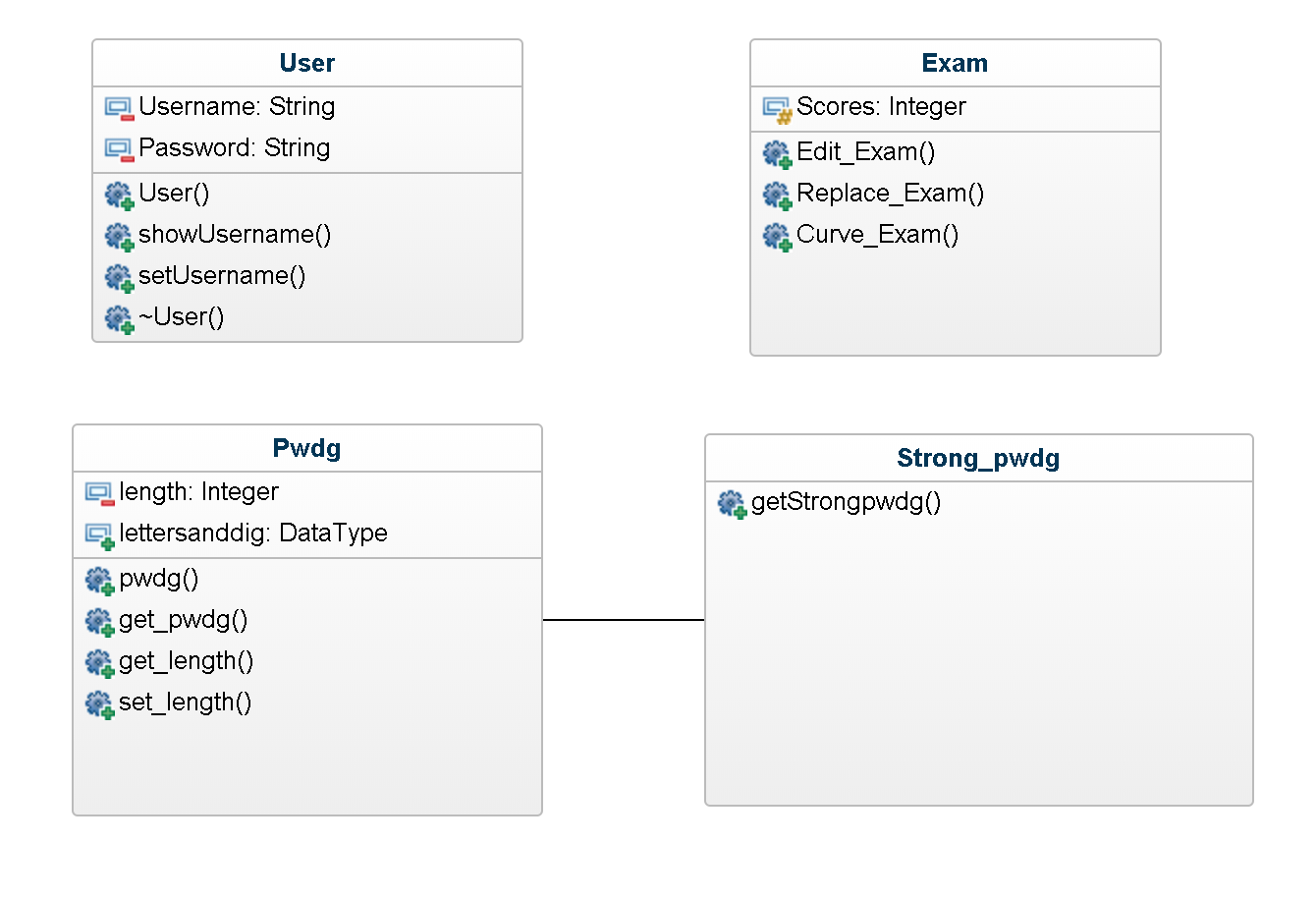
* report.docx (this report)

The runproject.bat simply runs main.exe (start main.exe in cmd) since I have put all source code into the main.cpp file.

The classes used in the program are below:

* User
* Exam
* Pwdg
* Strong\_pwdg

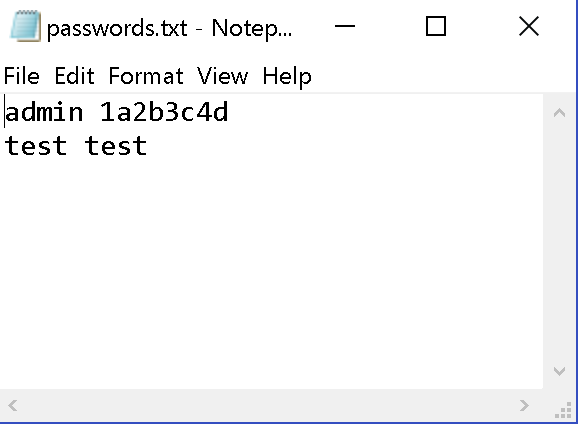
**The UML class diagram**

****

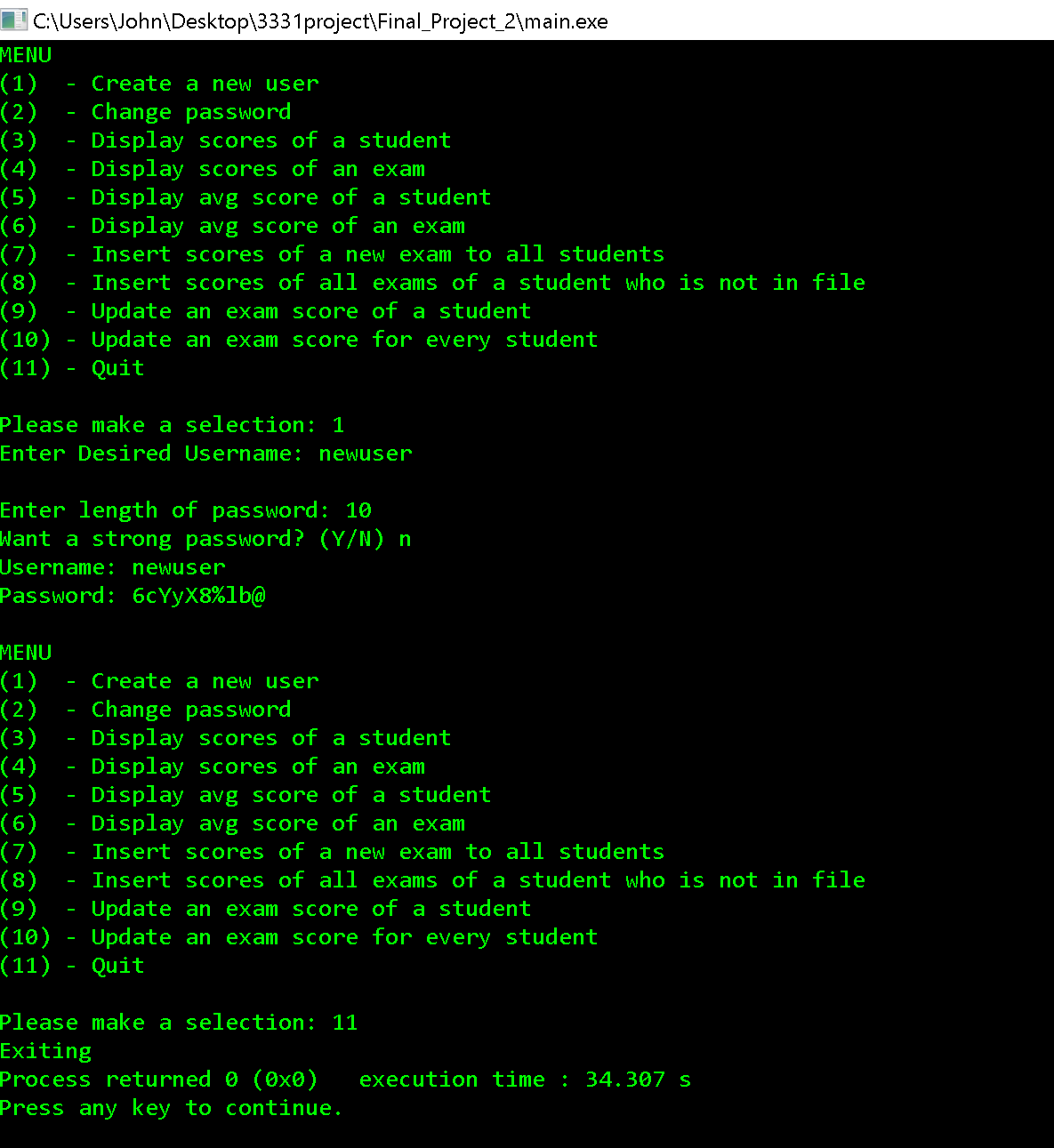
**Screenshots of menu items 1-11**

1. Create new user

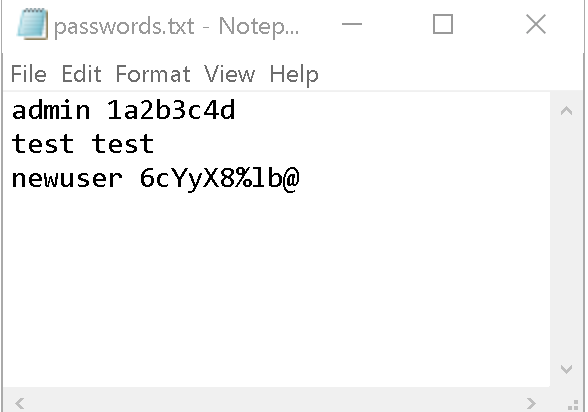
Passwords.txt file before



Program execution:



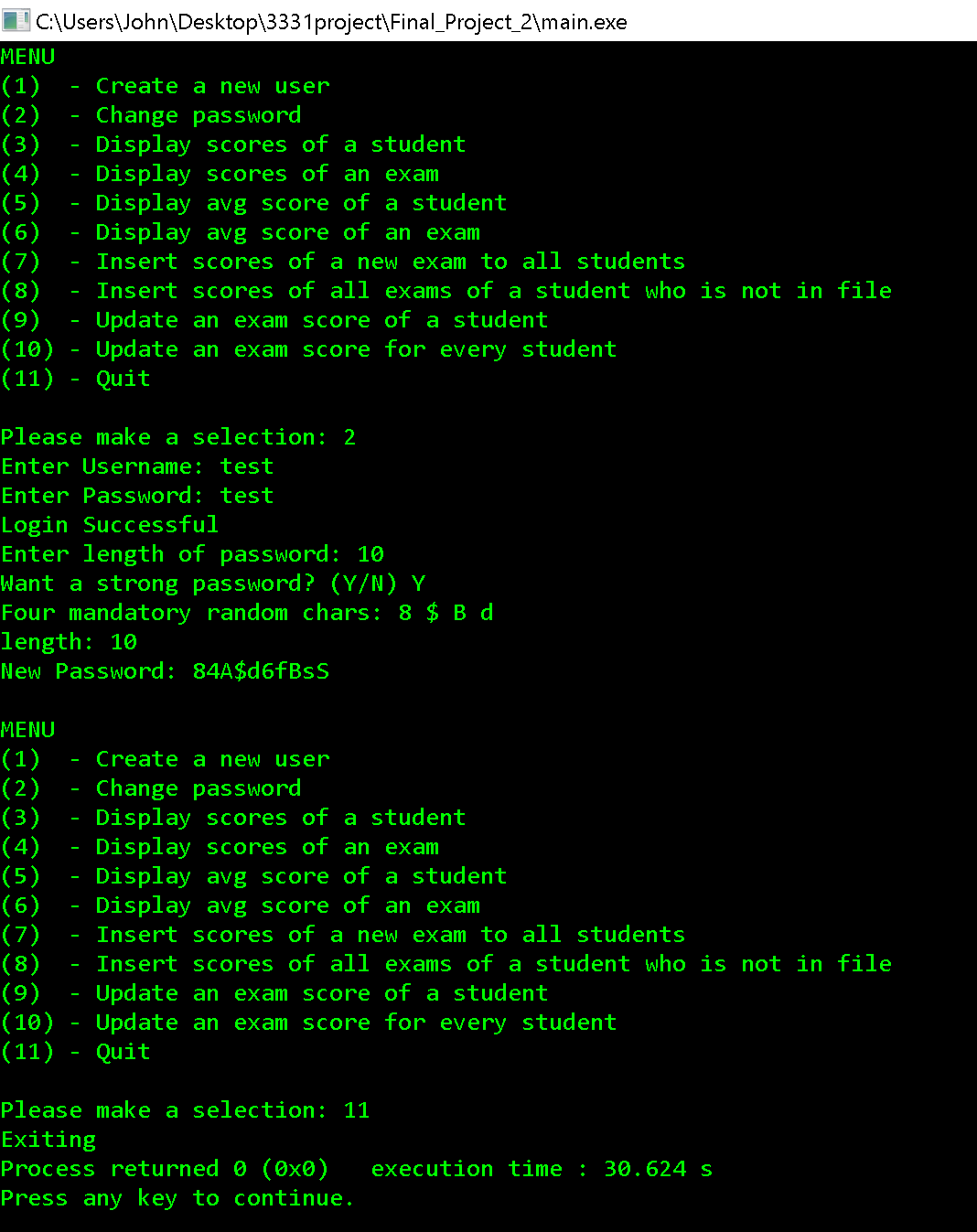
**Passwords.txt after execution:**



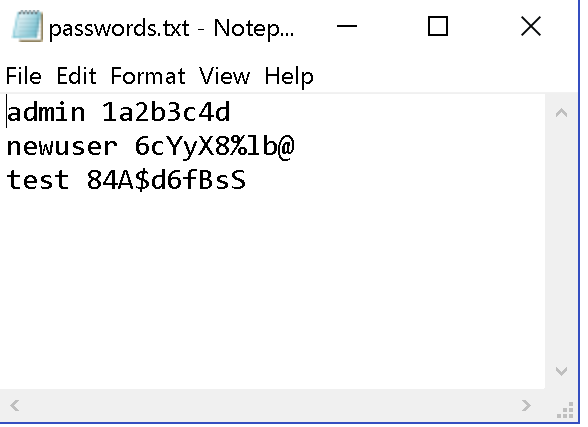
1. Change Password

passwords.txt is the same as picture above

Program execution(next page):



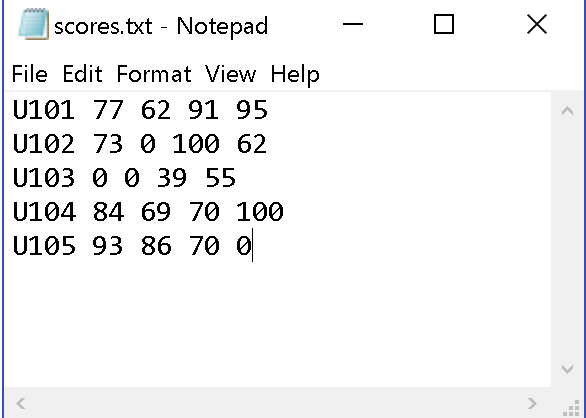
password.txt after program execution:



**Explanation**: you can see the password of user ‘test’ was changed and appended to end of text file

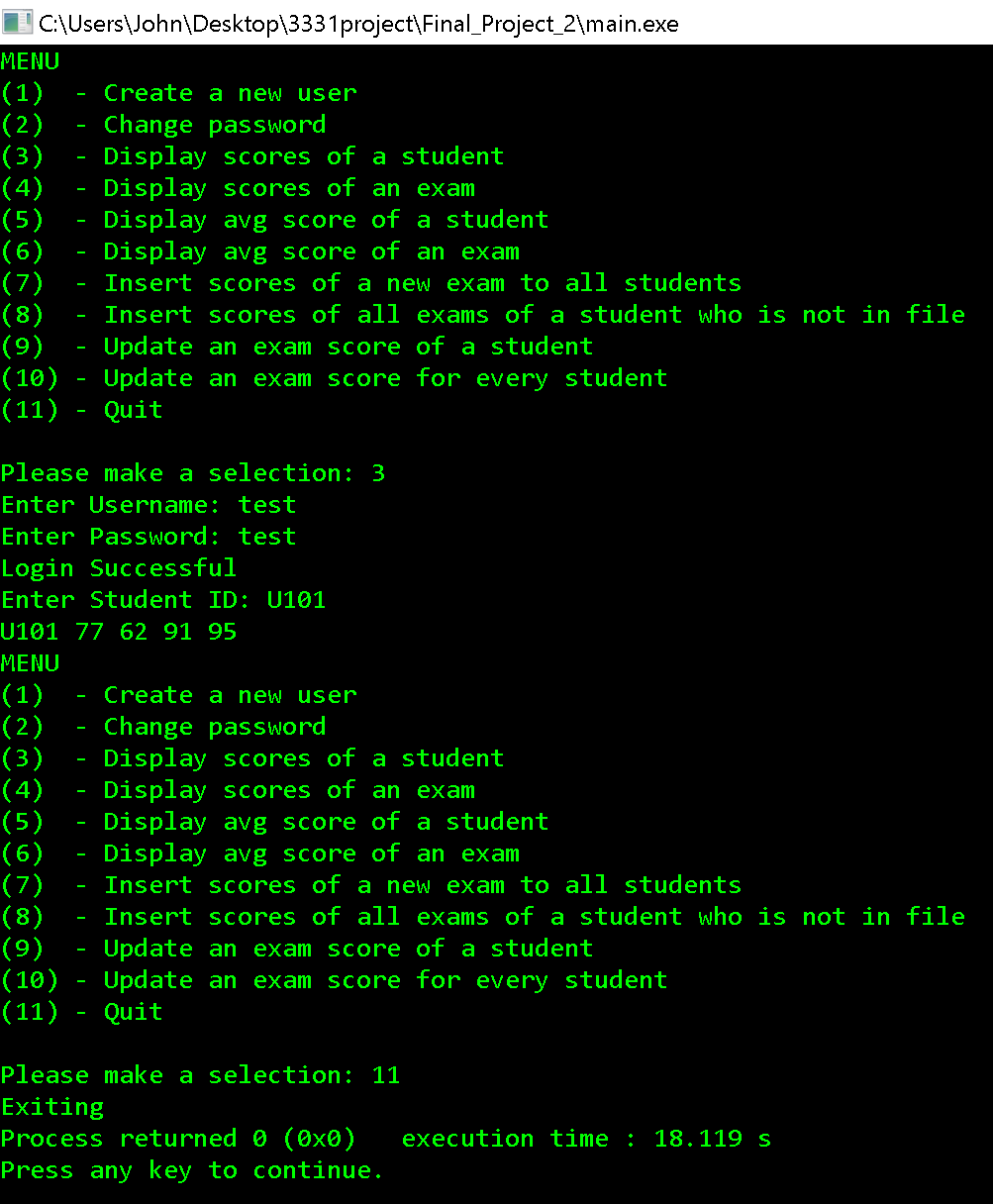
1. Display scores of a student

Scores.txt before program execution:



This is here to show the scores are read correct.

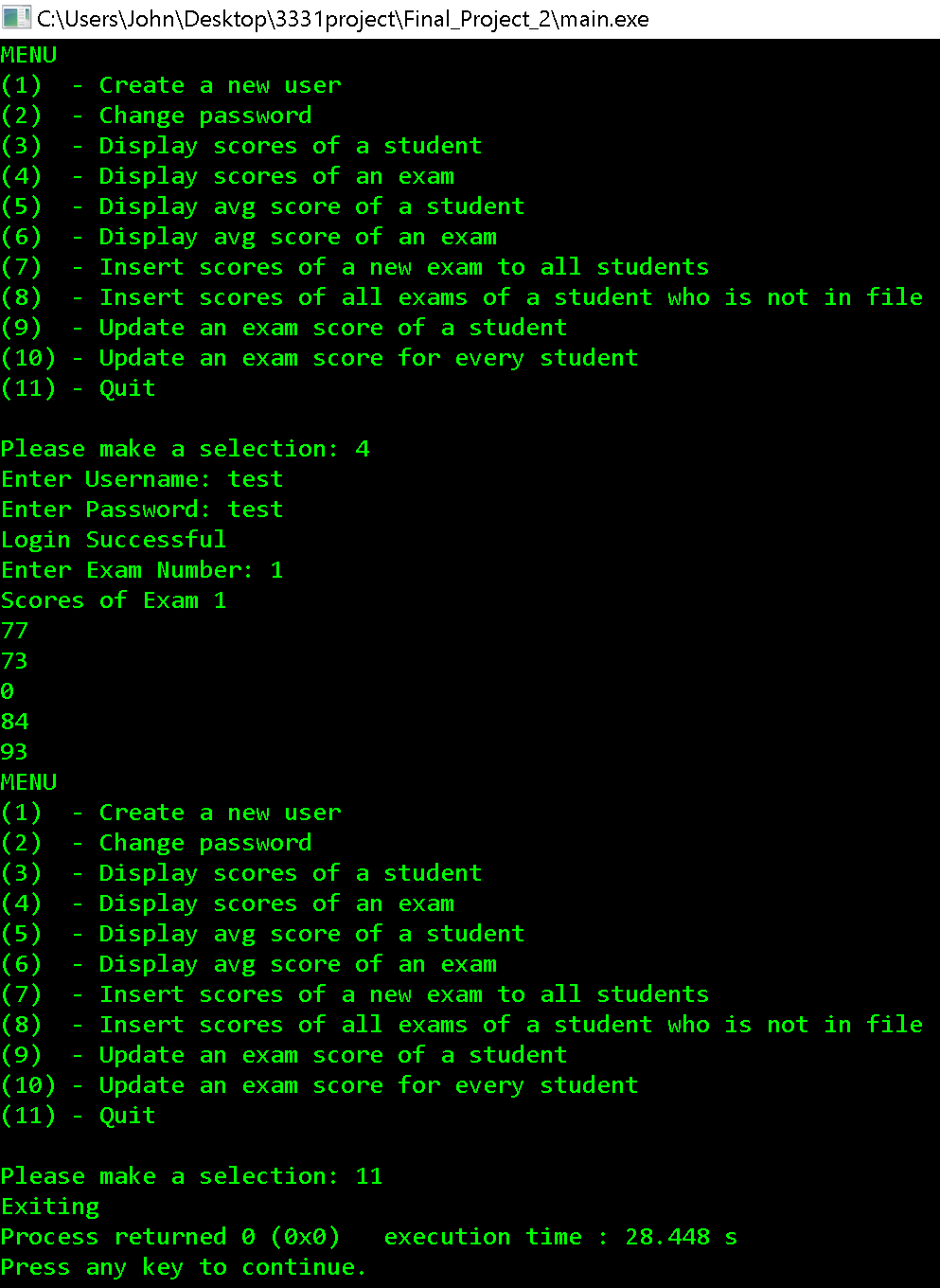
Program execution (next page):



**Explanation:** The first student was read from the test file and displayed.

1. Display Scores of an exam:

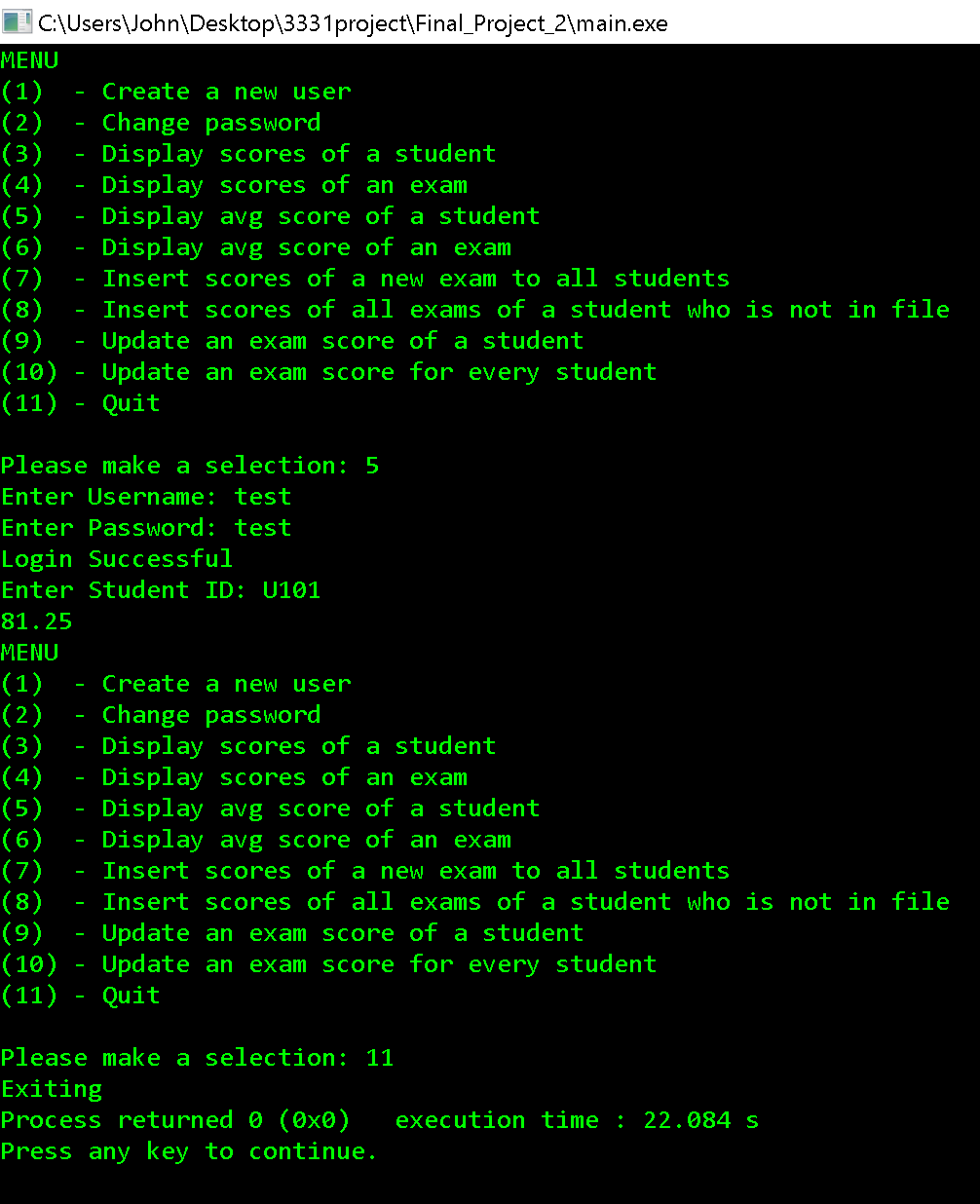
Program execution:



**Explanation**: The first column (exam) was read from the text file and displayed.

1. Display Average of a student

Program execution:

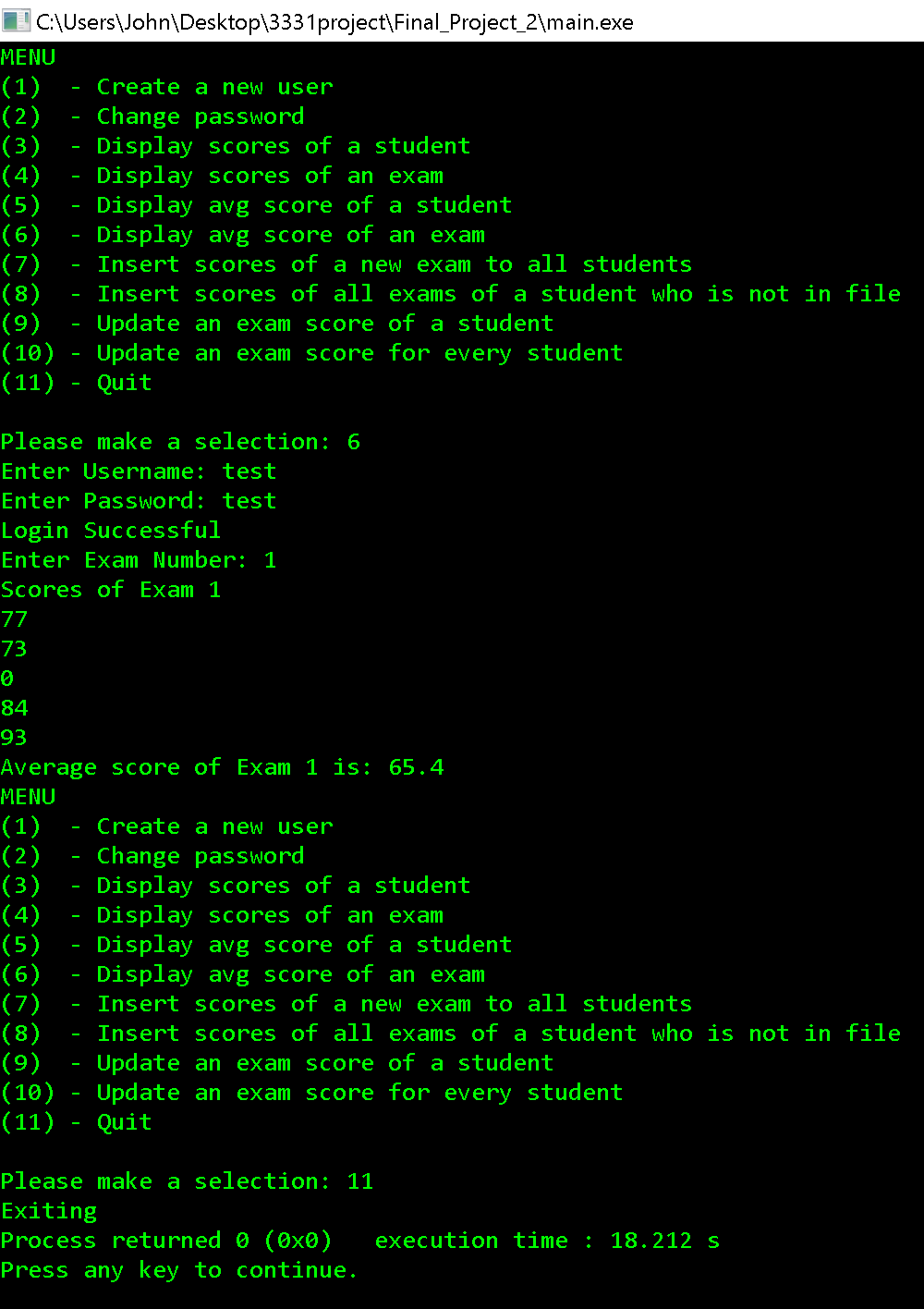


**Explanation**:

The first row was read from the text summed and averaged. (77 + 62 + 91 + 95) / 4 = 81.25

1. Display average of exam

Program Execution:

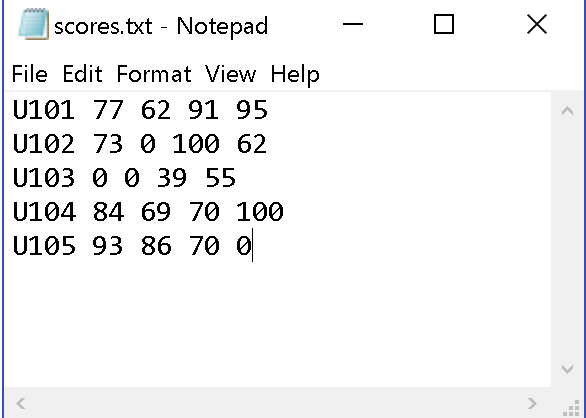


Explanation: The first column (exam) was read from the text file summed and averaged.

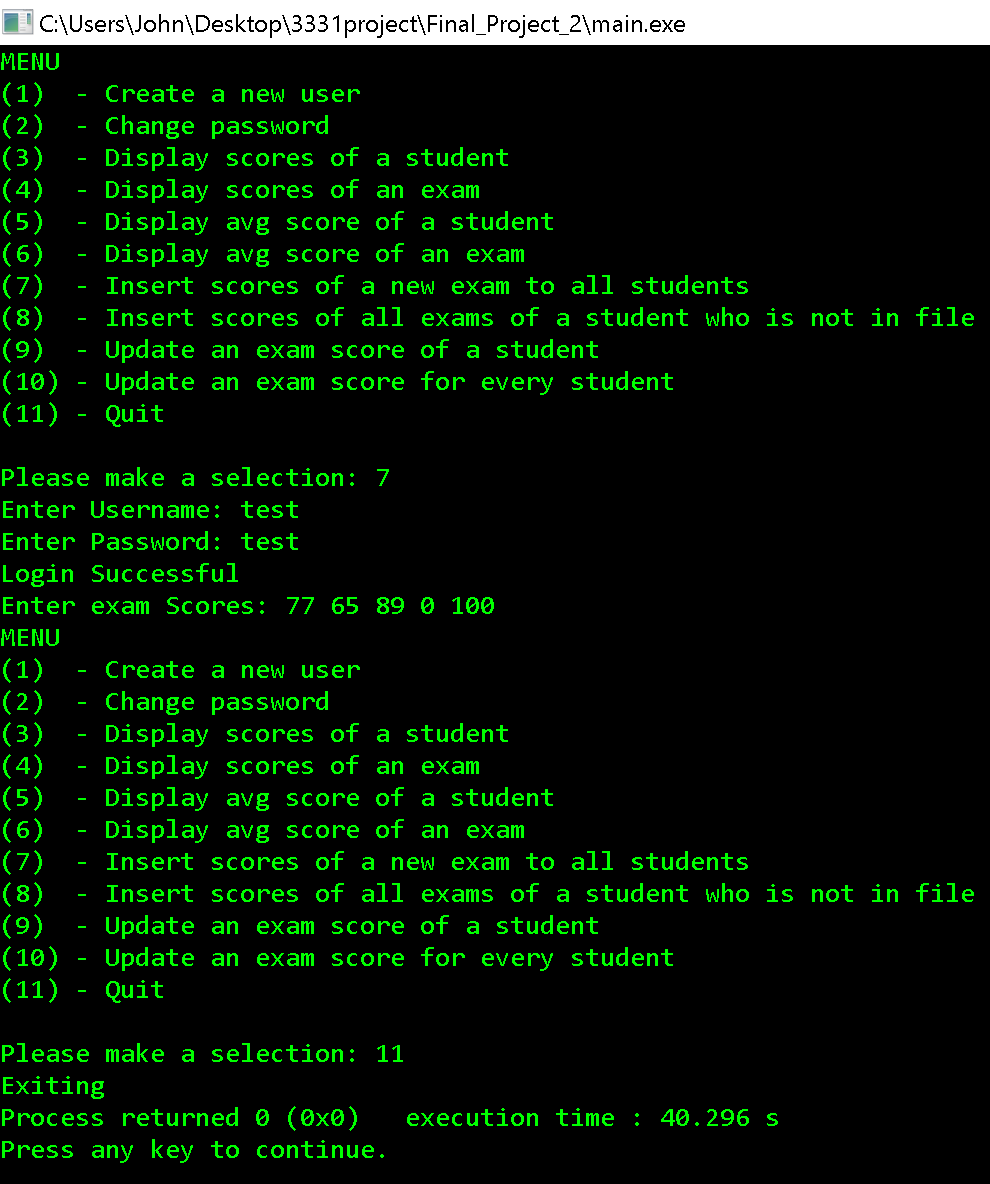
(77+73+0+84+93) / 5 = 65.4

1. Insert scores of a new exam to all students

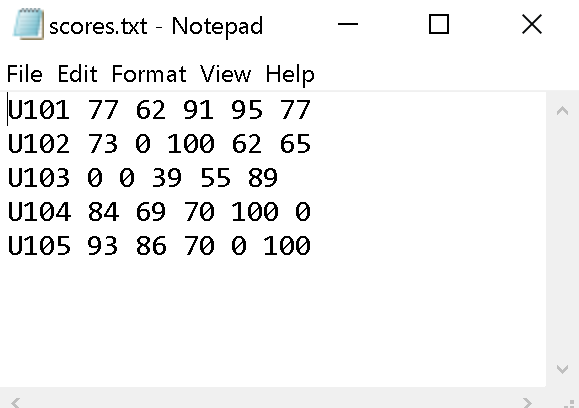
scores.txt before program execution:



Program Execution:



Scores.txt after execution:

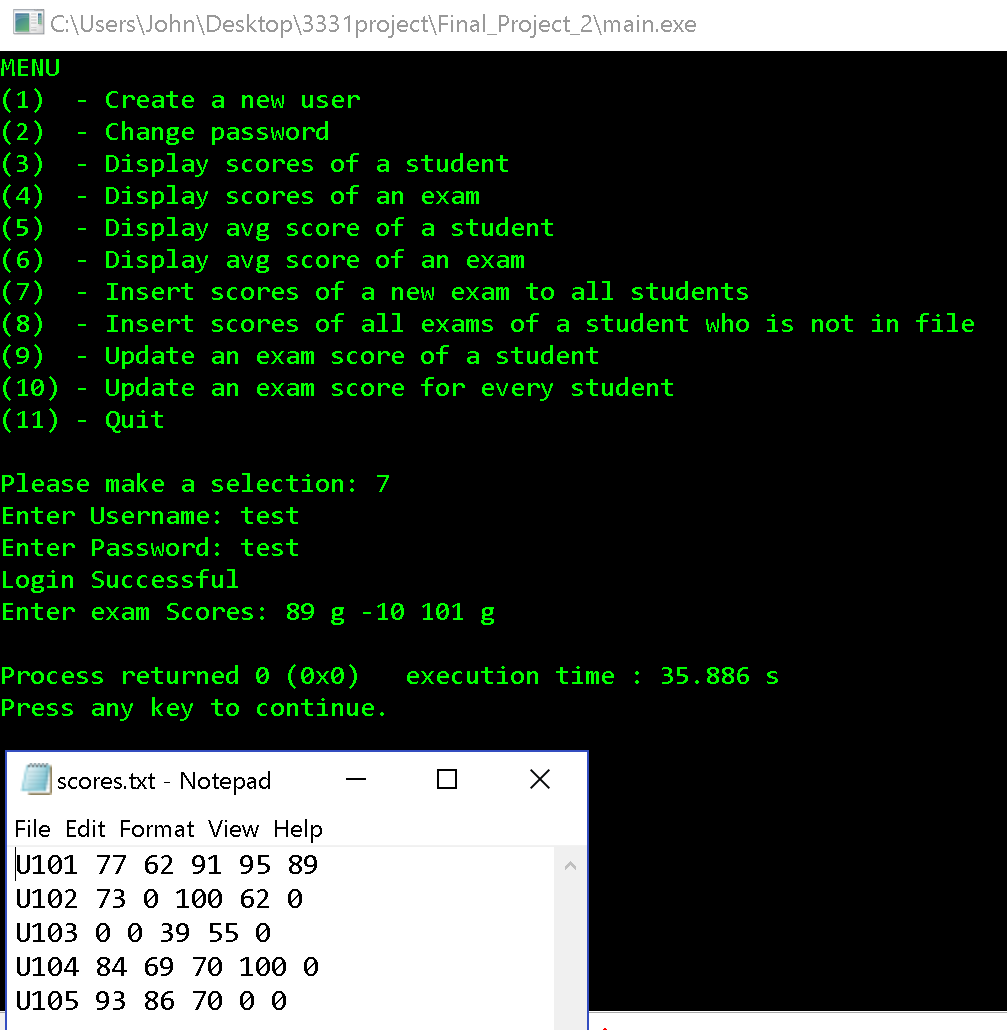


**Explanation**: As you can see from the picture above the exam scores were added to the end of each row for each student.

**Error Handling**:

If the user enters non-numbers/negative numbers/numbers>100, zeros will be written to the file for those entries.

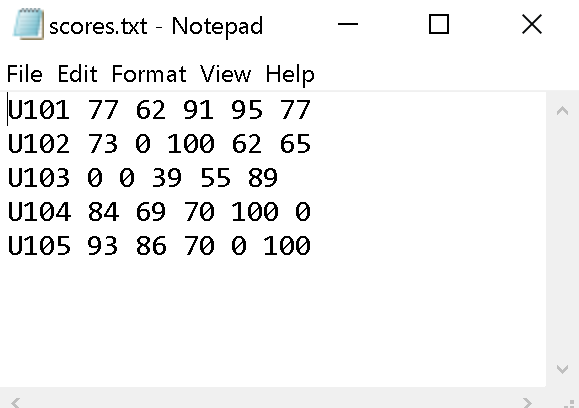
**Screenshots(next page):**

****

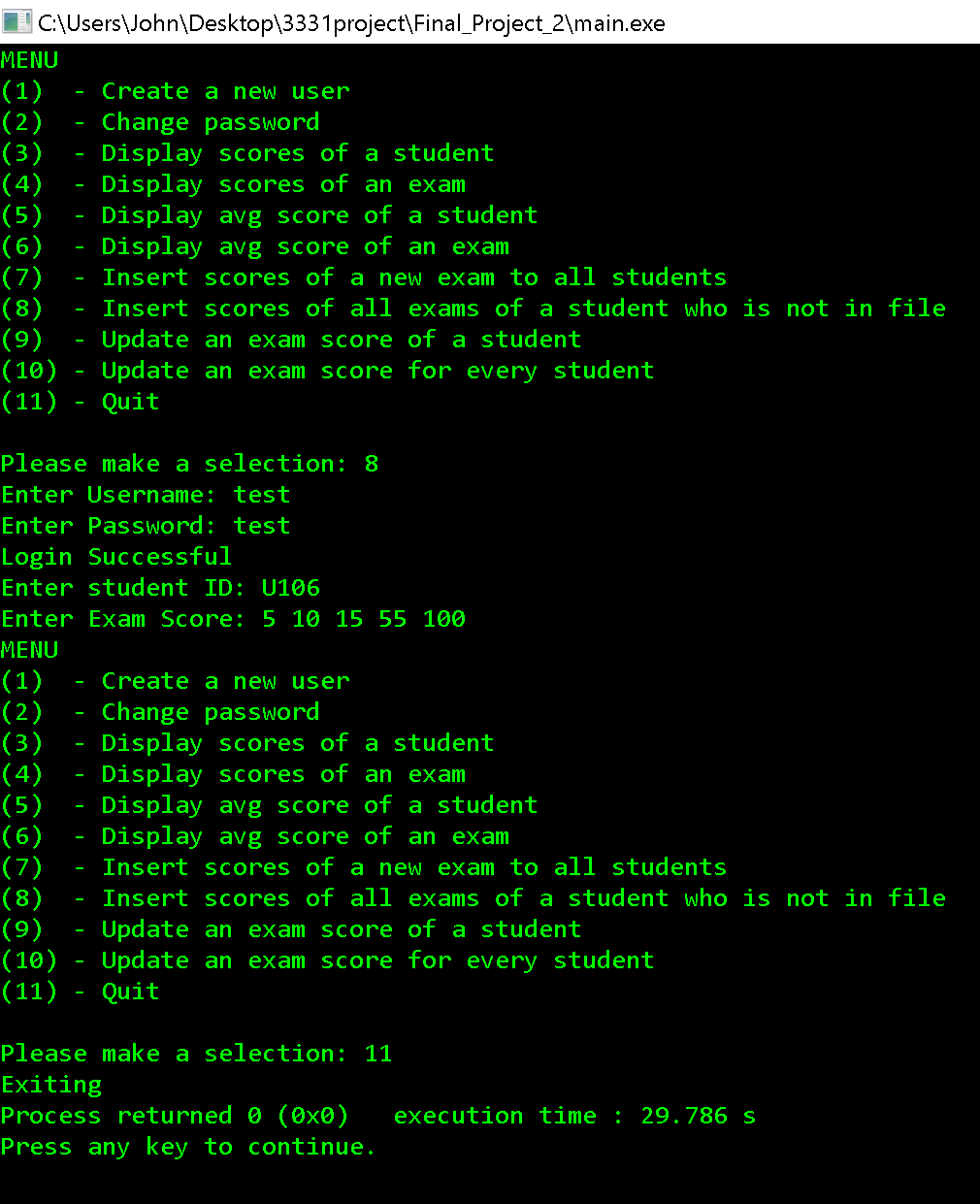
**Explanation**: Correct values written to column invalid written as zeros.

1. Insert scores of all exams of a student not on file (add a student and insert scores)

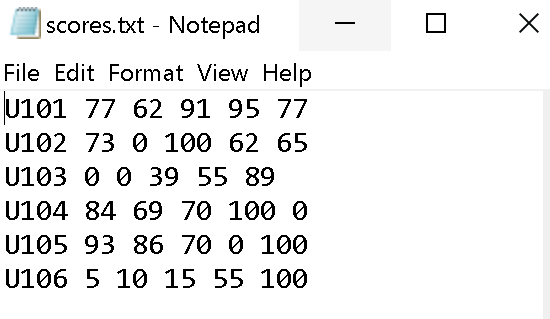
Scores.txt before execution:



Program Execution:



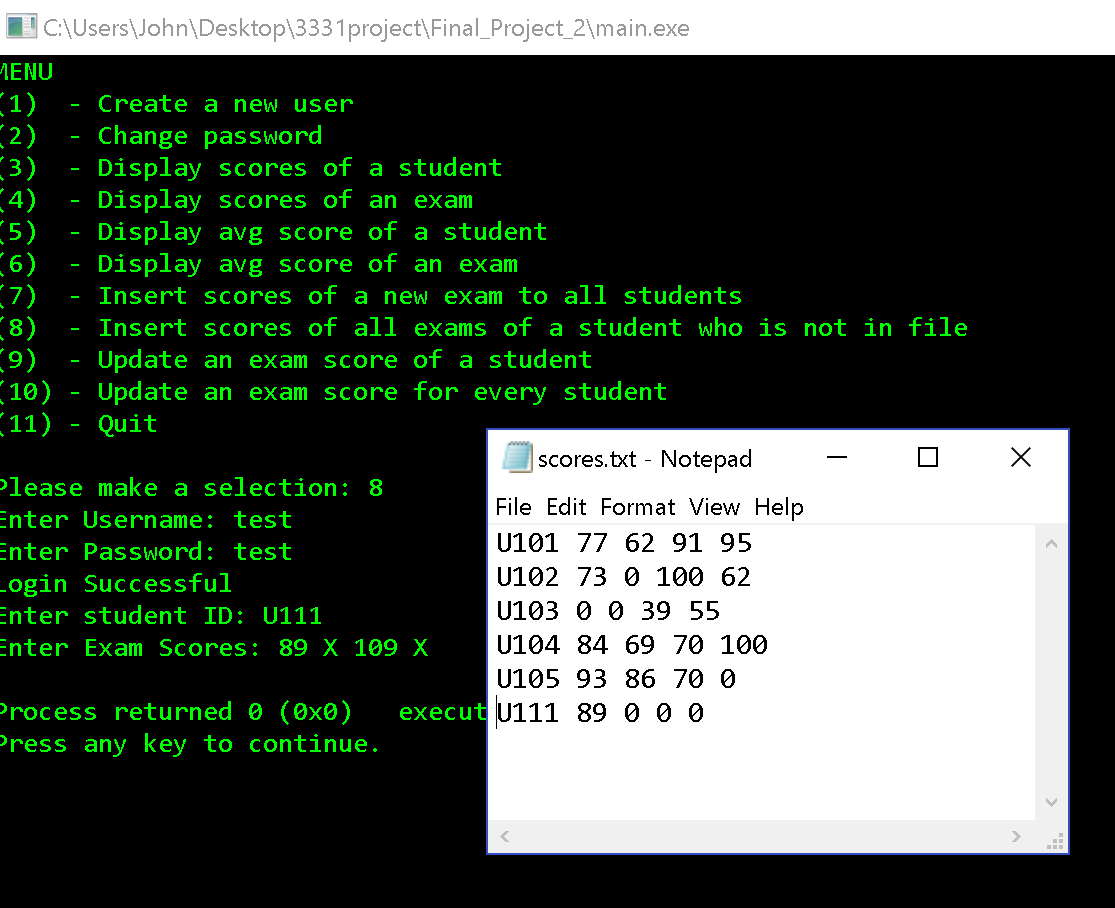
Scores.txt after:



**Explanation:** Student and scores appended to new row in text file.

**Error Handling:** If the user enters non-numbers/negative numbers/numbers>100, zeros will be written to the file for those entries.

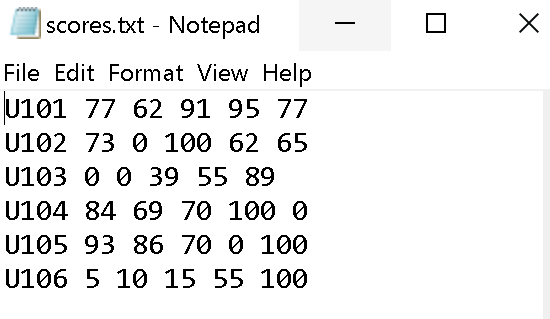
**Screenshots:**

****

**Explanation:** Invalid scores written as zero to file.

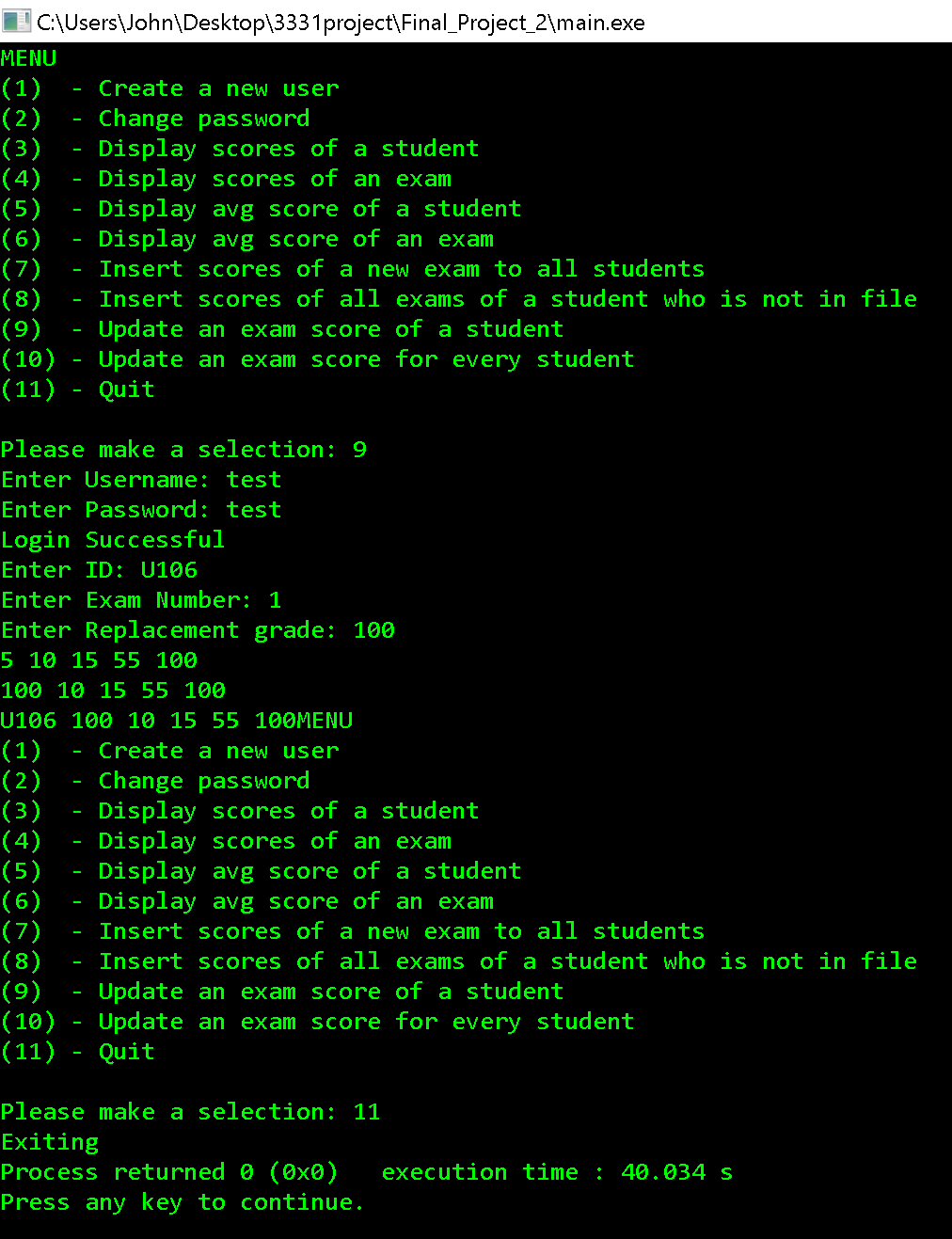
1. Update an Exam score of a student

Scores.txt before:

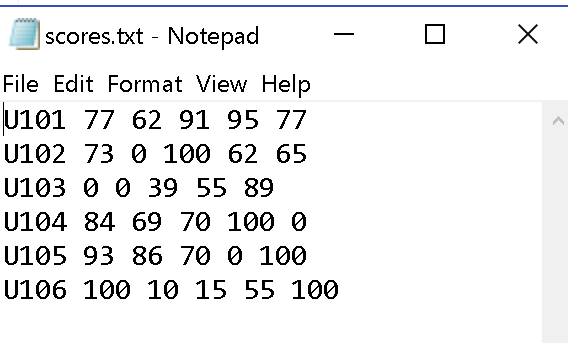


**Explanation**: We will be changing exam 1 of student U106 from 5 to 100.

**Program Execution (next page)**:



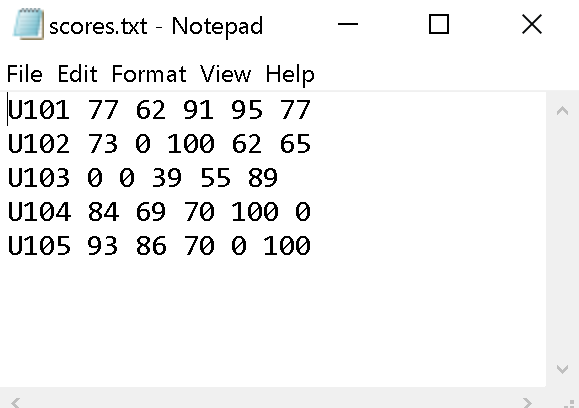
Scores.txt after execution:



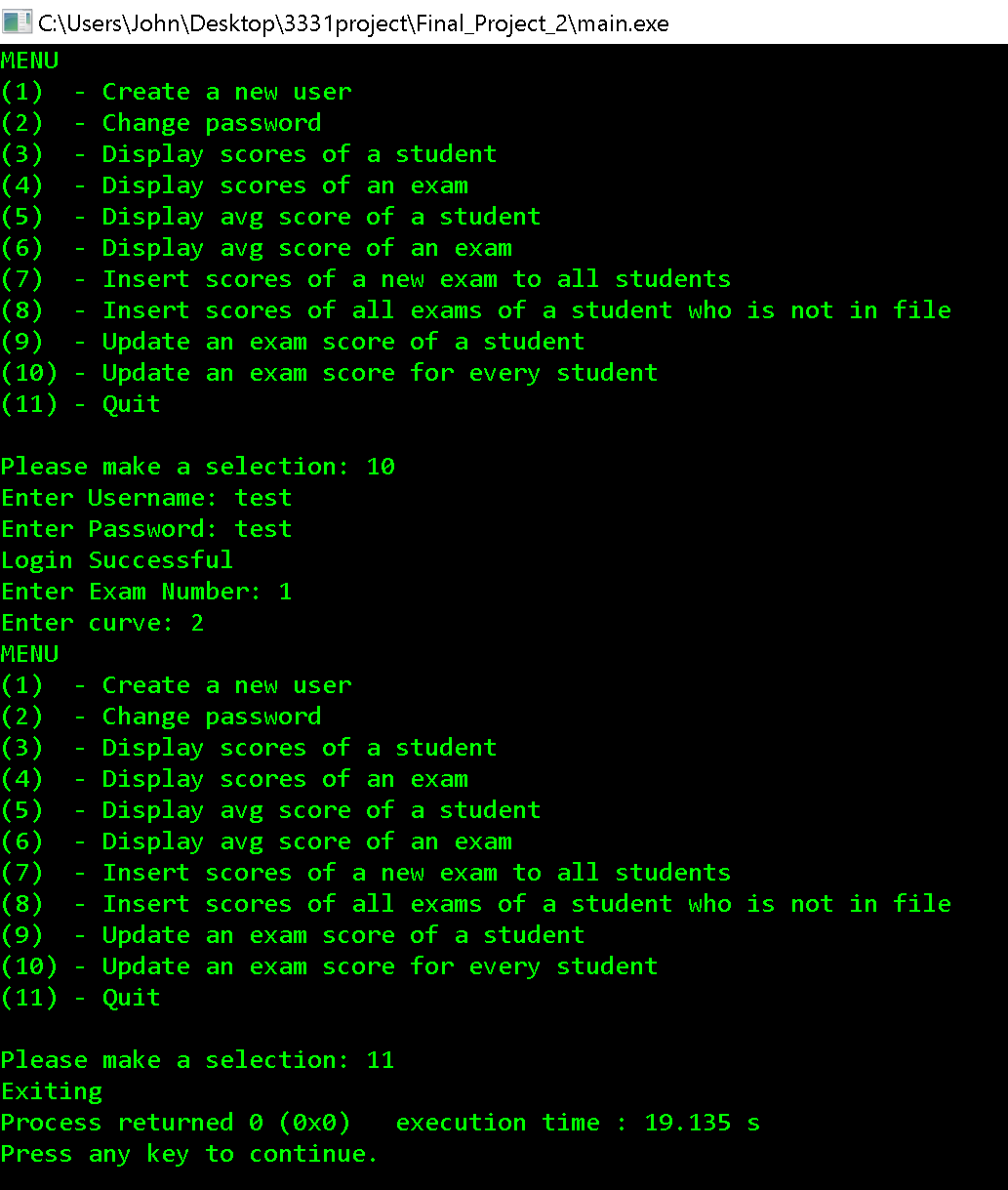
Explanation: Exam 1 of student U106 was successfully changed from 5 to 100.

1. Update an exam score for every student

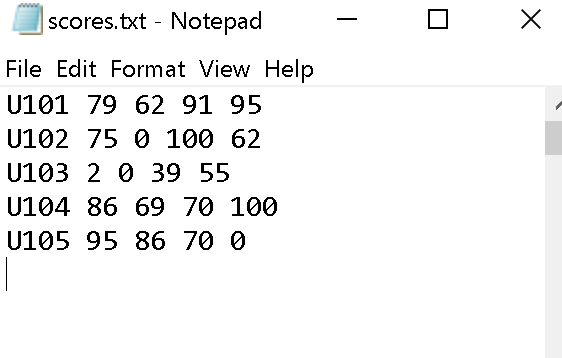
Scores.txt before:



**Program Execution**:



**After**:



**Explanation**: Curve added to each exam in first column.

**Conclusion**: If you have any issue with my program or concerns you can contact me at **johnadams@mail.usf.edu**