Greenville Technical College • CPT Department

CPT168: Programming Logic and Design

Lab Rules

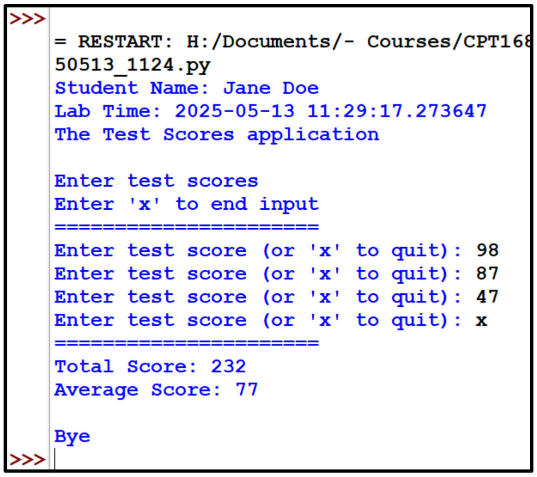
1. **<u>Lab File Naming Convention</u>**
    - Student's in CPT168 should use the following naming convention when submitting their lab assigments:
      Lastname_Firstname_LabCHP-NUM_name_of_the_Lab_as instructed_in_the_textbook.py
    - Example, for Lab 10-2, here is the correct file name for the file to be submitted in Blackboard by me:
      `Sanders_Beau_Lab10-2_enhanced_word_counter.py`
    - Data files should follow the naming convention in the textbook and restated in each lab in Blackboard with no other changes.
    - There will be point deductions for labs submitted with incorrect file names

2. **<u>Documentation Guidelines</u>**
    - **Follow these instructions for all coding labs in CPT168**; students not following all of these instructions for documentation will have points deducted from non-compliant lab submissions; documenting programming code files is an important part of learning how to become a programmer
    - Beginning with Lab01-3, all lab program and module files (.py) must be correctly documented
    - **Program and module files (.py pronounced “pie”) must have the following documentation**:
        - The **student’s name, date submitted, course-section number, and lab number** in the comments at the beginning of the file; for example:
          `#!/usr/bin/env python3` [this is a required line used to define the interpreter]
          `# Beau Sanders, May 13, 2025, CPT168-W01, Lab01-3`
        - After the line containing the student’s name there should be a comment (starts with #) **explaining what the program or module file is designed to do**
        - All functions and blocks of code should be preceded with a comment line (starts with #) to **explain or define what the function or block of code is designed to accomplish**; students are expected to document the code provided by the textbook publisher; do not assume the comments in the textbook provided files are sufficient, students should write their own documentation
        - **Data files (.txt, .csv, and .bin) should NOT have any documentation comments lines** because that could result in causing the related program or module to not run properly

3. **<u>Lab Output Screen Snip with Student’s Name and Run Date/Time</u>**
    - Students in CPT168 are required to add the following code at the beginning of each coding lab, above the header for the program; replace Jane Due with your first and last names:

```
import datetime
print(f"Student Name: Jane Doe")
labtime = datetime.datetime.now()
print(f"Lab Time: {labtime}")
```

- Students are required to submit a screen snip of their lab code output in order to get credit for the lab
- An example of a student's lab with the required code is at the end of this document.
- Be sure to replace "Jane Doe" with your first name and last name.
- Be sure to capture the beginning and ending prompts in IDLE.
- Here is an example of a good output screen snip for the example lab in this document; notice that the IDLE prompts at the beginning and end of the output are shown which is required:

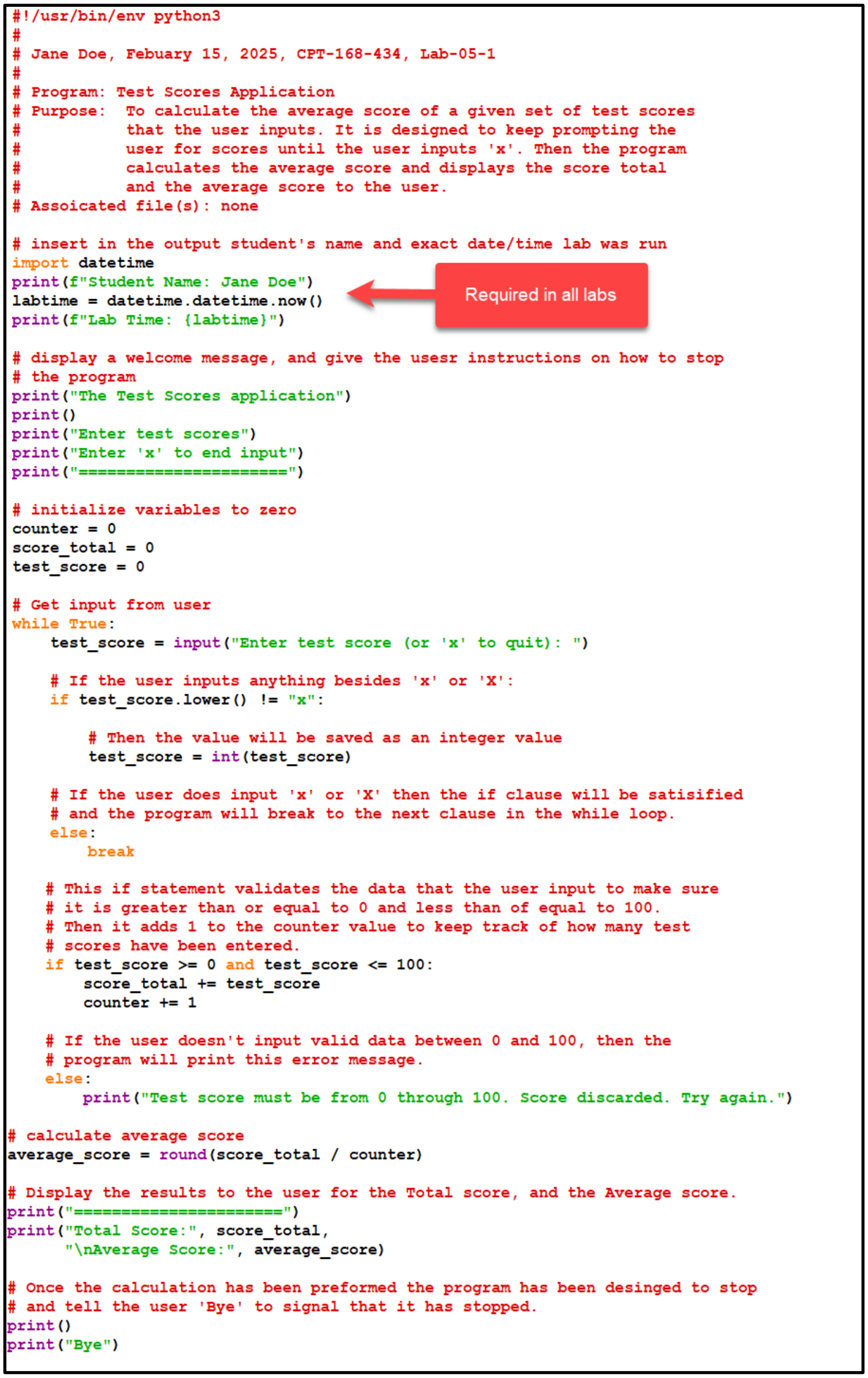


```
>>>
    = RESTART: H:/Documents/- Courses/CPT168
    50513_1124.py
    Student Name: Jane Doe
    Lab Time: 2025-05-13 11:29:17.273647
    The Test Scores application

    Enter test scores
    Enter 'x' to end input
    ======================
    Enter test score (or 'x' to quit): 98
    Enter test score (or 'x' to quit): 87
    Enter test score (or 'x' to quit): 47
    Enter test score (or 'x' to quit): x
    ======================
    Total Score: 232
    Average Score: 77

    Bye
>>>
```

4. Screen Snipping Tool Suggestions

- Students in CPT168 may use any screen snipping tool they prefer. Students will have to submit their CPT168 lab output as a **PNG image file** Here are three suggestions for snipping tools.
- **Snip and Sketch in Windows 10 and 11:** This tool is the more advanced version of the Snipping Tool from Windows 7. Snip and Sketch is a free tool and is well documented on the internet. Here is a video that explains how it works: https://youtu.be/OO0aS7gh3Rg. Students should use the default rectangle mode and save the command output lines including the command prompts at the beginning and end of the command as a PNG image file. Do NOT save and submit the whole screen. Only submit the output lines as specified in the Blackboard lab instructions. Follow the directions below.
- **Snipping Tool in Windows 10:** This is the original Snipping Tool that was introduced years ago. It works great and is free in Windows 10. The Windows Snipping Tool is also well documented on the internet. Here is another YouTube video that shows you how it works: https://youtu.be/5sphoFKPQDw. Students should save the command output lines including the command prompts at the beginning and end of the command as a PNG image file. Do NOT save and submit the whole screen. Only submit the output lines as specified in the Blackboard lab instructions. Follow the directions below.
- **Snagit from TechSmith:** This is a full featured screen capturing tool that creates images and videos. This tool is NOT free. Your instructor has been using it for years. Snagit is suggested, but NOT required. For more information go to this link: https://www.techsmith.com/screen-capture.html.

- **Example of a lab file that is well documented and shows the student's full understanding of the code**:



```
#!/usr/bin/env python3
#
# Jane Doe, Febuary 15, 2025, CPT-168-434, Lab-05-1
#
# Program: Test Scores Application
# Purpose:  To calculate the average score of a given set of test scores
#           that the user inputs. It is designed to keep prompting the
#           user for scores until the user inputs 'x'. Then the program
#           calculates the average score and displays the score total
#           and the average score to the user.
# Assoicated file(s): none

# insert in the output student's name and exact date/time lab was run
import datetime
print(f"Student Name: Jane Doe")
labtime = datetime.datetime.now()
print(f"Lab Time: {labtime}")

# display a welcome message, and give the usesr instructions on how to stop
# the program
print("The Test Scores application")
print()
print("Enter test scores")
print("Enter 'x' to end input")
print("======================")

# initialize variables to zero
counter = 0
score_total = 0
test_score = 0

# Get input from user
while True:
    test_score = input("Enter test score (or 'x' to quit): ")

    # If the user inputs anything besides 'x' or 'X':
    if test_score.lower() != "x":

        # Then the value will be saved as an integer value
        test_score = int(test_score)

    # If the user does input 'x' or 'X' then the if clause will be satisified
    # and the program will break to the next clause in the while loop.
    else:
        break

    # This if statement validates the data that the user input to make sure
    # it is greater than or equal to 0 and less than of equal to 100.
    # Then it adds 1 to the counter value to keep track of how many test
    # scores have been entered.
    if test_score >= 0 and test_score <= 100:
        score_total += test_score
        counter += 1

    # If the user doesn't input valid data between 0 and 100, then the
    # program will print this error message.
    else:
        print("Test score must be from 0 through 100. Score discarded. Try again.")

# calculate average score
average_score = round(score_total / counter)

# Display the results to the user for the Total score, and the Average score.
print("======================")
print("Total Score:", score_total,
      "\nAverage Score:", average_score)

# Once the calculation has been preformed the program has been desinged to stop
# and tell the user 'Bye' to signal that it has stopped.
print()
print("Bye")
```

Required in all labs

- **Video tutorial about documenting CPT168 lab coding files** (requires CPT168 shared password - cpt168/function): https://beausanders.org/CPT168/content/videos/How_to_Document_CPT168_Labs_220831_1052.mp4

5. **Data Files and Modules Needed for Labs To Run Properly**
    - **IMPORTANT**: When submitting labs, it is each CPT168 student's responsibility to include everything that is needed to grade the labs. This includes any data files (.txt, .csv, and .bin) created by your programs or read by your programs. This also includes module files (.py). If a student does not include the required data files, they will be graded accordingly.

6. **No ZIP Files**
    - CPT168 students will NOT be submitting any lab exercises in compressed ZIP files. If a student needs to submit multiple lab files, then attach each file separately to the Blackboard lab assignment.

Please email your instructor if you have any questions about CPT168 Labs and how to submit them.

-Beau Sanders - sandewbs@my.gvltec.edu
CPT168 Course Lead Instructor

-Hazel Tucker - tuckehbt@my.gvltec.edu
CPT168 Instructor

-Adam Richards - richatar@my.gvltec.edu
CPT168 Instructor

Last Updated: May 15, 2025