

**CSCI 1301 – Programming Principles I**  
**Georgia Southern University**  
**Department of Computer Science**  
**Fall 2024**

**Assignment 5**

**Point Value: 20 points**

**Due: Friday October 4, 2024, start of lab**

**NOTE:** Gradescope will give input test cases for any problems that require it. Use the Scanner's `nextDouble()`, `nextInt()`, etc. methods discussed in class to prompt the user for input and Gradescope will act as the user when your program requests it. In other words, just code user input like normal—as if a human were going to test your program.

**Description**

Write two variants of the same Java program. Only one submission checklist should be turned in for this assignment.

**PAssign05a (10 points) – while/do-while loops**

Write a Java program using a while or do-while loop that reads in exam grades from the user until a sentinel is entered. Once the user is done entering grades the program should display the number of valid grades entered, number of grades rejected, the lowest grade, and highest grade entered. Additionally, classify the grades and count how many As, Bs, Cs, Ds, and Fs there were, and display them in your final output.

Grades should be in the range from 0 to 100 and your program should ensure that invalid grades are noted; an error message should be displayed, and program execution should keep executing, but the grades should not be included in the count or used for the lowest/highest grade determination. The sentinel should not be counted as one of the grades entered or used for lowest/highest grade determination.

**PAssign05b (10 points) – for loops**

Write a Java program that asks the user how many exam grades they will be entering and, using a for loop, prompts the user for that many grades. Once the user is done entering grades the program should display the number of valid grades entered, number of grades rejected, the lowest grade, and highest grade entered. Additionally, classify the grades and count how many As, Bs, Cs, Ds, and Fs there were, and display them in your final output.

Grades should be in the range from 0 to 100 and your program should ensure that invalid grades are noted; an error message should be displayed, and program execution should keep executing, but the grades should not be included in the count or used for the lowest/highest grade determination.

## NOTES

- For both programs, if no valid grades are entered, make sure the lowest grade is displayed as 0.00. If valid grades are entered, make sure the actual, correct lowest grade is recorded.
- Do not mix token-based and line-based String input. Remember the problems that occur when mixing the two.
- Use a single tab character (\t) to create the column-based output including number of letter grades).

## Expected Output/Sample Runs (Do NOT just hardcode output)

<p>Enter grade from 0-100, -999 to stop: 900  Error: That is not a valid score.  Enter grade from 0-100, -999 to stop: 100  Enter grade from 0-100, -999 to stop: 78  Enter grade from 0-100, -999 to stop: -999</p> <p>Valid grades:           2  Invalid grades:        1  Highest grade:        100.00  Lowest grade:         78.00</p> <p>As:     1  Bs:     0  Cs:     1  Ds:     0  Fs:     0</p>	<p>Number of Grades: 3  Enter grade from 0-100 (1 of 3): 900  Error: That is not a valid score.  Enter grade from 0-100 (2 of 3): 100  Enter grade from 0-100 (3 of 3): 78</p> <p>Valid grades:           2  Invalid grades:        1  Highest grade:        100.00  Lowest grade:         78.00</p> <p>As:     1  Bs:     0  Cs:     1  Ds:     0  Fs:     0</p>
<p>Enter grade from 0-100, -999 to stop: -100  Error: That is not a valid score.  Enter grade from 0-100, -999 to stop: 101  Error: That is not a valid score.  Enter grade from 0-100, -999 to stop: -50  Error: That is not a valid score.  Enter grade from 0-100, -999 to stop: 102  Error: That is not a valid score.  Enter grade from 0-100, -999 to stop: -999</p> <p>Valid grades:           0  Invalid grades:        4  Highest grade:        0.00  Lowest grade:         0.00</p> <p>As:     0  Bs:     0  Cs:     0  Ds:     0  Fs:     0</p>	<p>Number of Grades: 4  Enter grade from 0-100 (1 of 4): -100  Error: That is not a valid score.  Enter grade from 0-100 (2 of 4): 101  Error: That is not a valid score.  Enter grade from 0-100 (3 of 4): -50  Error: That is not a valid score.  Enter grade from 0-100 (4 of 4): 102  Error: That is not a valid score.</p> <p>Valid grades:           0  Invalid grades:        4  Highest grade:        0.00  Lowest grade:         0.00</p> <p>As:     0  Bs:     0  Cs:     0  Ds:     0  Fs:     0</p>
PAssign05a – while/do-while Loop version (2 runs)	PAssign05b – for Loop version (2 runs)

## Code

Use the provided template for this assignment. Make any necessary modifications to classes and class headers to complete this assignment.

```
public class PAssign05 { // add a or b as necessary
    public static void main(String[] args) {
        // add your code here
    }
}
```

### **Deliverables**

Name your program PAssign05.java. Programming Assignment 5 is to be individual work. Submit the program by the specified due date. Submit each file to its corresponding assignment on Gradescope.

See and follow the Programming Assignment Format document for submission requirements.

Use a utility similar to <https://www.diffchecker.com/> and the Expected Output to compare your program's output with the requested output as well as the unit tests provided within Gradescope. Programming is in the details, so double check punctuation, spacing, and case if your output does not match. When copying and pasting, be aware that Microsoft Word sometimes replaces normal quotes with Smart Quotes, which may need to be edited.

Last modified: September 23, 2024