CMSC 131 Quiz 2 Worksheet

The second quiz for the course will be on Thursday, Feb. 25. The following list provides additional information about the quiz:

- The quiz will be posted on Thursday, Feb. 25, 8 AM (morning), and due the same day, Thursday, Feb. 25, at 4 PM (afternoon).

- It is designed to be completed in less than 1 hour, but I am making it available for 8 hours since people have different schedules.

- You will not have lab on Thursday Feb. 25, so that should free up 1 hour for everyone to work on the quiz.

- We will have normal office hours on Thursday Feb. 25, but TAs cannot answer any questions about the quiz in OH (They can help you submit if you have submit server issues).

- Did you install the correct version of Eclipse, Java 15, and course management software on your computer at the start of the semester? See here: http://www.cs.umd.edu/eclipse/install/

If you don't have this exact setup and you are not able to submit the quiz, that will not be a valid reason for an extension.

- The quiz will be posted similar to a class project. You will write code in an Eclipse project and submit as usual.

- You can only post clarification questions in Piazza on quiz day and a CMSC 131 staff member will reply. **You should post as a private post and we will make it public or update the FAQ if others can benefit from the answer. As a student, do not answer any piazza post on quiz day**. Debugging questions, why code is not compiling, why is code not passing a test, are invalid questions to post in Piazza.

- Posting of any kind of code in Piazza (or other public platforms), during the quiz period, represents an academic integrity violation and will be reported as such.

- The quiz will be graded based on submit server tests (release and secret) and code inspection (e.g. style, following rules, etc.). The exact rubric will not be available before the quiz. Just follow all the rules to avoid point deductions

- **You must work by yourself.** Sharing of quiz solutions represents an academic integrity violation and will be reported as such. Submissions can be checked with cheating detection software.

- You can use class resources (lecture notes, lecture/lab examples, videos, etc.), but no other resources (e.g., code from the web).

- All submissions must be done via the submit server (no e-mail). The highest scoring submission on the submit server will be downloaded for manual TA grading purposes (you can submit as many times as you want before the deadline).

- There will be a 1-hour late submission period, therefore you need to submit often and before Thursday, Feb. 25, at 4 PM (afternoon) for your quiz to count on time. If you turn it in between 4 and 5 PM, it will be marked late and there will be a 5-point deduction. Questions will not be answered on piazza during the late period.

- If you are student with an extended time accommodation from ADS, the time frame provided takes into consideration your time allocation. If you need any other assistance or still have concerns to finish the quiz, contact me via email before the quiz day.

- The quiz will cover concepts covered in lecture and lab during Week 1 to Week 3 (The main focus will be looping. Static methods will not be on the quiz).

- It is in your best interest to complete this work by yourself, and following the guidelines provided above. You need to identify which topics you understand and which ones you don't, so you can be successful in CMSC132 and future CS courses. The following exercises gives you practice with concepts that may show up on the quiz. Solutions to these exercises will not be provided, but you are welcome to discuss your solutions with the TAs during office hours or on Piazza.

Exercises

1. When do we use a do while?
2. When do we use a for loop?
3. Write a program that reads two integer values and prints the even numbers in that range.
4. Write a program that computes the factorial of a number. For example, factorial of 4 (4!) is 24.
5. Write a program that prints the sum of odd numbers between min (inclusive) and max (inclusive). The program will read the values min and max and display the sum. You can assume min and max are integer values. Use the Scanner class to read values.

6. Write a program that reads (using the Scanner class) an integer value representing the number of rows associated with a triangular diagram. After reading the value, your program will generate a triangular diagram with the specified number of rows. For example, if the user enters 4 your program will print:

```
   *
  **
 ***
****
```

7. Write a program that prints a triangle where each row has twice as many characters as the preceding one. The first row will have 2 characters. The program will read the size (an integer value) of the triangle and a character. It will then generate a triangle with a number of rows that corresponds to size and where the character provided is used for odd-numbered rows. Even-numbered rows will always use a *. We use the message “Enter size:” and “Enter character:” to read data. Use the Scanner class to read data. Below we have provided two examples of running the program. Notice your program must work for other values. Underlined text represent input provided by the user.

```
Enter size: 3
Enter character: $
$$
****
$$$$$$$$
```

```
Enter size: 4
Enter character: #
##
****
########
****************
```

Below is an actual quiz from Fall 2019. I am just making this old quiz available for further practice, but remember the format of your quiz in Spring 2021 will be different than what we did in the past. Therefore, to make sure you do well on the quiz you need to be comfortable with **the material covered in our class**, not just the solution to this sample quiz. You will find our quiz much more difficult *if you only study this quiz* that was designed to be done in 10 minutes with no book, notes, or computer.

- http://www.cs.umd.edu/class/spring2021/cmsc131-01XX/quizzes/Quiz2_Fall_2019.pdf
- http://www.cs.umd.edu/class/spring2021/cmsc131-01XX/quizzes/Quiz2_Fall_2019_Soln.pdf