

## CMSC 131 Quiz 1 Worksheet

The first quiz for the course will be on Thursday, Feb. 11. The following list provides additional information about the quiz:

- The quiz will be posted on Thursday, Feb. 11, 8 AM (morning), and due the same day, Wed, Feb. 11, 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. 11, so that should free up 1 hour for everyone to work on the quiz.
- We will have normal office hours on Thursday, Feb. 11, 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 Thurs, Feb. 11, 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 and Week 2 (Looping 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. You need to be familiar with the following examples of academic integrity violations:

- Hardcoding of results in a project assignment. Hardcoding refers to attempting to make a program appear as if it works correctly (e.g., printing expected results for a test).
- Using any code available on the internet/web or any other source. For example, using code from Sourceforge.
- Hiring any online service to complete an assignment for you.
- Sharing your code or your student tests with any student.
- Using online forums (other than Piazza) in order to ask for help regarding our assignments.

2. You need to be familiar with information provided at:

<http://www.cs.umd.edu/class/resources/academicIntegrity.html>

3. What is the good faith attempt for a project? Would you fail the class if you don't satisfy a good faith attempt? See syllabus for information regarding good faith attempts.

4. Which of the following are valid (will compile) Java identifiers?

**House house #dog cat% blue-elephant 12depth \$height**

5. What is bytecode? If we compile a class called **Ex.java**, what is the name of the file with bytecode?

6. Order the following types based on their relative size.

**short int byte**

7. What is the default type (float or double) for the value 4.5?

8. What is an appropriate name (using camel case) for a class representing a computer?

**Computer computer COMPUTER**

9. What is an appropriate name (using camel case) for a variable representing tire pressure?

**tirePressure TirePressure tire\_Pressure tire\_pressure TIRE\_PRESSURE TIRE\_Pressure**

10. Write a Java Program that displays the following message:

**The directory for Thomas "tommy" project is \fs\www\myproject**

11. Define a symbolic constant that represents PI (3.14).

12. Write a program that asks the user for a password value; the expected value is "terps". If the user provides the expected value, the program will print the message "Access Granted"; otherwise the program will print the message "Access Denied."

13. Complete the program below. The program reads an integer value and prints the square if the value is greater than 0; otherwise the message "Invalid value" will be printed. Notice the quotes must surround the words Invalid value. For example, if the user enters 5, the program will print 25.

```
public class Square {  
    public static void main(String[] args) {
```

14. Complete the program below that computes tuition cost. The program will read the cost per credit (in dollars) using the message "Enter cost per credit: " and the number of credits using the message "Enter number of credits: ". If the number of credits is larger than or equal to 18, the cost per credit will be reduced by half. The program will display the amount to pay using the message "Please pay: " followed by the amount. Remember that the Scanner methods `nextDouble()` and `nextInt()` allow you to read a double and an integer, respectively. The following are examples of running the program. Underlined text represent input provided by the user.

Enter cost per credit: <u>30.50</u> Enter number of credits: <u>2</u> Please pay: 61.0	Enter cost per credit: <u>2</u> Enter number of credits: <u>18</u> Please pay: 18.0
--	---

```
public class ComputeTuition {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);
```

Below are actual quizzes from Spring 2019 and Fall 2019. I am just making these old quizzes 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 solutions to these sample quizzes. You will find our quiz much more difficult *if you only study these quizzes* that were designed to be done in 20 minutes with no book, notes, or computer.

- <http://www.cs.umd.edu/class/spring2021/cmssc131-01XX/quizzes/Quiz1131Spring19.pdf>
- <http://www.cs.umd.edu/class/spring2021/cmssc131-01XX/quizzes/Quiz1131Spring19Soln.pdf>
- <http://www.cs.umd.edu/class/spring2021/cmssc131-01XX/quizzes/Quiz1131Fall19.pdf>
- <http://www.cs.umd.edu/class/spring2021/cmssc131-01XX/quizzes/Quiz1131Fall19Soln.pdf>