COP 3330 Quiz #1 Solutions

1/30/2026

1) (10 pts) Jamila is making fruit baskets which have both apples and oranges. Each basket will have the same number of apples and the same number of oranges. Complete the main method below that it reads in the number of apples and oranges needed for each basket as well as the total number of apples and oranges that she has, and outputs the total number of complete fruit baskets she can form. *(For full credit please use a method from the Math library and no if statement. If you don't you'll lose 1 point.)* You may assume that each value entered by the user will be a positive integer.

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

    int res = 0;
    Scanner stdin = new Scanner(System.in);
    System.out.println("How many apples per basket?");

    int applesPerBasket = stdin.nextInt();              // Grading: 1 pt

    System.out.println("How many oranges per basket?");

    int orangesPerBasket = stdin.nextInt();             // Grading: 1 pt

    System.out.println("How many apples do you have total?");

    int applesTotal = stdin.nextInt();                  // Grading: 1 pt

    System.out.println("How many oranges do you have total?");

    int orangesTotal = stdin.nextInt();                 // Grading: 1 pt

    int appleLimit = applesTotal/applesPerBasket;       // Grading: 2 pts
    int orangeLimit = orangesTotal/orangesPerBasket;    // Grading: 2 pts
    res = Math.min( appleLimit, orangeLimit );          // Grading: 2 pts

    System.out.println("You can make a total of "+res+" complete baskets.");
}
```

2) (5 pts) Write a segment of code that creates one Random object and prints out 100,000 randomly generated integers, each in between -1,000,000 and 1,000,000, inclusive.

```java
Random rndObj = new Random();                       // Grading: 1 pt
for (int i=0; i<100000; i++)                         // Grading: 1 pt

    // Grading: 1 pt print, 1 pt -1000000, 1 pt rnd call
    //          Give 2 out of 3 if off by 1 error
    System.out.println(-1000000+rndObj.nextInt(2000001));
```

3) (6 pts) Of each item listed below, clearly circle each one that is a primitive type in Java:

number   **long**   **char**   String   **boolean**   int[]

Grading: 1 pt for each item (You get the point if you correctly circle or DON'T circle it.)

4) (4 pts) From memory, list at least one method from the Math library **that isn't listed on the reference sheet at the end of this exam.** List the name of the method and its formal parameter list and give a brief description of what it does.

There are many: anything but abs, max, min and pow will do. Here's one:
```
// Returns the square root of its argument a, provided a>=0,
// Returns NaN if a is negative.
static double sqrt(double a);
```

Grading: 1 pt method name, 1 pt parameter list (just types required), 2 pts explanation of what it does, partial up to the grader.

5) (10 pts) Here is an inverted pyramid of size 5:

```
5 5 5 5 5
 4 4 4 4
  3 3 3
   2 2
    1
```

Complete the main method below, using a nested loop structure, so that it prints out an inverted pyramid of size n, where n is guaranteed to be a positive integer in between 1 and 9, inclusive. All of your print statements can print out at most two characters.

```
public static void main(String[] args) {
    Scanner stdin = new Scanner(System.in);
    System.out.println("How many rows for your inverted pyramid?");

    int n = stdin.nextInt();                          // Grading: 1 pt

    for (int i=n; i>=1; i--) {                         // Grading: 2 pts

        for (int j=n; j>i; j--)                        // Grading: 2 pts
            System.out.print(" ");                     // Grading: 1 pt

        for (int j=0; j<i; j++)                        // Grading: 2 pts
            System.out.print(i+" ");                   // Grading: 1 pt

        System.out.println();                          // Grading: 1 pt
    }
}
```

Grading Note: There are many, many ways to do this, map the points conceptually for different solutions. Just take off 1 pt total if they always println or never println.

6) (9 pts) Jessica does different workouts each day, repeating the same pattern every 10 days. If we label the days 0, 1, 2, ..., 9, the days 0, 3, 6 and 9, are "leg day"s, days 4 and 8 are "arm day"s and the rest of the days in the cycle are "cardio". Complete the program below **using a SINGLE comparison in two of the branches** so that the program prints out what work out will occur on each day starting from day 0 ending at day $n - 1$, where n is entered by the user. (Note: day 10 will be a leg day, day 11 a cardio day, day 12 cardio, day 13 leg, day 14 arm day, etc.)

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

    Scanner stdin = new Scanner(System.in);
    System.out.println("How many days of working out?");
    int n = stdin.nextInt();

    for (int i=0; i<n; i++) {
        System.out.print("Day "+i+": ");

        if ( (i%10) %3 == 0)                        // Grading: 3 pts

            System.out.println("leg day");          // Grading: 1 pt

        else if ( (i%10) %4 == 0)                   // Grading: 3 pts

            System.out.println("arm day");          // Grading: 1 pt
        else

            System.out.println("cardio");           // Grading: 1 pt
    }
}
```

Grading Note: Order is forced so students don't get the print point if it's not in the right slot.

7) (5 pts) All four women's semifinalists in the Australian Open have both their first and last names end in the letter a (Aryna Sabalenka, Elina Svitolina, Jessica Pegula and Elena Rybakina). Complete the segment of code below, assuming that the first name of a player is stored in the String variable first and the last name of the same player is stored in the String variable last, that prints out "A+" if both names end in a either 'a' or 'A', and prints out "not a semifinalist" otherwise. **Please use the appropriate String methods and do not check for 'a' and 'A' separately.**

```java
Scanner stdin = new Scanner(System.in);
System.out.println("Please enter your first and last name.");
String first = stdin.next();
String last = stdin.next();

int len1 = first.length();
int len2 = last.length();

if (first.toLowerCase().charAt(len1-1) == 'a' &&
     last.toLowerCase().charAt(len2-1) == 'a')
    System.out.println("A+");
else
    System.out.println("not a semifinalist");
```

Grading: 1 pt if-else structure with 2 prints, 1 pt toLowerCase or toUpperCase calls, 2 pts charAt calls, 1 pt correct boolean expression

8) (1 pt) Artemis II is likely going to launch later this year, with four astronauts. What was its unmanned predecessor, which was launched on November 16, 2022?

Artemis I **(Grading: Give pt to all)**

Java Language Reference

Math Class

Modifier and Type	Method	Description
`static double`	`abs(double a)`	Returns the absolute value of a double value.
`static double`	`max(double a, double b)`	Returns the greater of two double values.
`static double`	`min(double a, double b)`	Returns the smaller of two double values.
`static double`	`pow(double a, double b)`	Returns the value of the first argument raise to the power of the second argument.

Random Class

Modifier and Type	Method	Description
`double`	`nextDouble()`	Returns the next pseudorandom, uniformly distributed double value between 0.0 and 1.0 from this random number generator's sequence.
`int`	`nextInt(int bound)`	Returns a pseudorandom, uniformly distributed int value between 0 (inclusive) and the specified value (exclusive), drawn from this random number generator's sequence.

String Class

Modifier and Type	Method	Description
`char`	`charAt()`	Returns the char value at the specified index.
`int`	`length()`	Returns the length of this string.
`String`	`toLowerCase()`	Converts all of the characters in this String to lower case using the rules of the default locale.
`String`	`toUpperCase()`	Converts all of the characters in this String to upper case using the rules of the default locale.