

COP 3330 Exam #1 9/21/2026 Solutions

Please print your name in CAPITAL letters.

1) (10 pts) Complete the program below so that the user enters the total distance of a road trip (in miles), the average speed (in miles per hour) when driving during the road trip and the number of minutes during the trip where the car was stopped (for rest or refueling) and prints out the total time, in hours, that the trip will take. **Each value entered by the user will be an integer. In order to earn full credit, you must store each value entered by the user in an integer variable. You may declare any other variables of any type as you need.**

```
public static void main(String[] args) {  
  
    double res = 0.0;  
    Scanner stdin = new Scanner(System.in);  
    System.out.println("How long was the trip in miles?");  
    int length = stdin.nextInt(); // 1 pt  
  
    System.out.println("What was the average speed in mph for the trip?");  
    int mpg = stdin.nextInt(); // 1 pt  
  
    System.out.println("How many minutes did you stop during the trip?");  
    int stopMin = stdin.nextInt(); // 1 pt  
                                     // 1 pt all 3 ints  
  
    // 6 pts breakdown below  
    res = 1.0*length/mpg + stopMin/60.0;  
  
    // 1 pt for assignment, 1 pt divide length mpg, 1 pt do double div,  
    // 1 pt add stop time, 1 pt divide stopMin by 60, 1 pt double div  
  
    System.out.println("Your trip took "+ res +" hours.");  
}
```

2) (8 pts) Write a segment of code that creates one Random object and prints out 100 randomly generated integers, in increasing order such that each subsequent integer is in between 1 and 100 more than the previous randomly generated integer, with each of these 100 choices being equally likely. (For example, if we've already generated 27, 50, 126, then the fourth integer must be in between 127 and 226, inclusive, with each choice equally likely.) The first integer should be in between 0 and 99, inclusive. **(Note: Do NOT use Math.random().)**

```
Random rndObj = new Random(); // 1 pt  
int cur = rndObj.nextInt(100); // 2 pts  
System.out.print(cur); // 1 pt  
for (int i=1; i<100; i++) { // 1 pt  
    cur += (rndObj.nextInt(100)+1); // 2 pts  
    System.out.print(" "+cur); // 1 pt  
}  
System.out.println(); // 0 pts
```

// Note: Don't take off for off by 1 errors.

3) (8 pts) Complete the method below that takes in a String str and a char ch, and prints out each index location in str that ch occurs. Please solve this task by only using the String method .charAt:

```
public static void printLocs(String str, char ch) {  
  
    for (int i=0; i<str.length(); i++)           // 3 pts  
        if (str.charAt(i) == ch)                 // 4 pts  
            System.out.print(i+" ");             // 1 pt  
    System.out.println();                         // 0 pts  
  
    // Give partial as you see fit.  
}
```

4) (8 pts) Now, solve the same problem given in question 3 but this time without using the charAt method and instead using both versions of the indexOf method listed at the end of the exam.

```
public static void printLocs(String str, char ch) {  
  
    int idx = 0;                                  // 1 pt  
    while (true) {                                // 1 pt  
        idx = str.indexOf(ch, idx);               // 3 pts  
        if (idx == -1) break;                     // 1 pt  
        System.out.print(idx+" ");               // 1 pt  
        idx++;                                    // 1 pt  
    }  
  
    // Note: Loop control is 2 pts total.  
}
```

5) (15 pts) Here is a layered square with three layers:

```
3 3 3 3 3
3 2 2 2 3
3 2 1 2 3
3 2 2 2 3
3 3 3 3 3
```

Complete the main method below, using a nested loop structure, so that it prints out a layered square 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. Note: Think carefully about the relationship between the loop indexes and the number being printed. Also, a method from the math library might be quite helpful.**

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

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

    for (int i=-n+1; i<n; i++) { // 2 pts
        for (int j=-n+1; j<n; j++) { // 2 pts

            int val = Math.max( Math.abs(i)+1, Math.abs(j)+1 ); // 6 pts
            System.out.print(val+" "); // 2 pts
        }

        System.out.println(); // 2 pts
    }
}

// 6 pt breakdown - 2 pts for max of two items
//                  2 pts for correct distance from center row
//                  2 pts for correct distance from center column
//                  give 1 pt for loops if they run the wrong number of times
```

6) (10 pts) The law of cosines allows you to solve for the length of the third side of a triangle if you know the lengths of the other two sides and the angle in between those two known sides. Specifically, if a and b are the known side lengths, and C is the included angle in between those sides, then we can solve for the third unknown side length, c, as follows:

$$c = \sqrt{a^2 + b^2 - 2ab\cos(C)}$$

Write a function that takes as input side lengths a and b as well as the included angle C (in radians), and returns the length of side c. **You must call the following methods from the math class: sqrt, pow, cos.**

```
public static double getSideC(double a, double b, double angleC) {  
  
    return Math.sqrt( Math.pow(a,2) + Math.pow(b,2) - 2*a*b*Math.cos(angleC) );  
  
    // 1 pt return, 1 pt sqrt, 2 pts 1st pow call, 2 pts second pow call,  
    // 2 pts cos call, 2 pts for +, - and parentheses in correct places  
}
```

7) (10 pts) What is the output of the following code segment?

```
String orig = "AbCDeFaBCDef";  
String second = orig.toLowerCase();  
String s3 = orig.replace("a", "Z");  
String s4 = second.replace("a", "Z");  
String s5 = second;  
second = second.toUpperCase();  
System.out.println(orig+", "+second);  
System.out.println(s3+", "+s4+", "+s5);
```

```
AbCDeFaBCDef, ABCDEFABCDEF  
AbCDeFZBCDef, ZbcdefZbcdef, abcdefabcdef
```

Grading: 2 pts per correct string, 0 pts otherwise (no partial on a string because the errors just create slight differences anyway.)

8) (10 pts) What is the output of running the main method shown below?

```
public static void main(String[] args) {  
    int[] vals = {3, 9, 12, 4, 1, 8, 11, 16, 7, 5};  
    int[] other = new int[10];  
    for (int i=0; i<10; i++)  
        other[i] = (vals[i] + vals[(i+1)%10])/2;  
    for (int i=0; i<10; i++)  
        System.out.print(other[i]+", ");  
    System.out.println();  
}
```

6, 10, 8, 2, 4, 9, 13, 11, 6, 4,

Grading: 1 pt per corresponding number, no exceptions.

9) (20 pts) A matrix is a two dimensional grid of numbers. Two matrices can be added if they have the same dimensions (same # of rows and same # of columns). To add two matrices, you just add the two corresponding values at each location. For example,

$$\begin{bmatrix} 3 & 4 & 2 \\ 7 & 1 & 6 \end{bmatrix} + \begin{bmatrix} 1 & 4 & 5 \\ 8 & 3 & 6 \end{bmatrix} = \begin{bmatrix} 4 & 8 & 7 \\ 15 & 4 & 12 \end{bmatrix}$$

On the next page, complete the method so that it adds the two input matrices and returns the resulting matrix as a two-dimensional array. If the dimensions of the two input matrices don't match, your method should return null. In your method, make sure to allocate space for a new two-dimensional array, store the appropriate values in that array and return that array.

```

public static int[][] add(int[][] m1, int[][] m2) {

    // 5 pts - 1 pt if, 1 pt first comparision, 2 pts second
    // comparison, 1 pt return
    if (m1.length != m2.length || m1[0].length != m2[0].length)
        return null;

    // 4 pts - 1 pt lhs, 1 pt new, 1 pt first size, 1 pt
    // second size
    int[][] res = new int[m1.length][m1[0].length];

    // 2 pts
    for (int i=0; i<m1.length; i++)

        // 2 pts
        for (int j=0; j<m1[0].length; j++)

            // 6 pts total - 1 pt LHS, 2 pts each term, 1 pt add
            res[i][j] = m1[i][j] + m2[i][j];

    // 1 pt
    return res;

}

```

10) (1 pt) Five Guys Burgers and Fries was started by Jerry Murrell and his wife, Janie in 1986. The five guys refers to both Jerry and his sons that were already born at the time. How many sons did Jerry have, back in 1986?

Four

Note: According to Wikipedia, the original five guys were Jerry and his four sons who were already born at the time. He had a fifth son two years after starting the company. All of the sons are still active in running the company today!