Practice Midterm 1 Answers

Chapters 1 - 3

**You have 50 minutes to work on this exam. Answer each question to the best of your ability. If you are stumped, or don’t know how to write a specific piece of code, use comments, or sudo code, to describe the process or steps necessary to writing the program. As a suggestion, make tables to keep track of the variables in questions one and two if necessary.**

**Good Luck ☺**

**1. What is the output of the following program?**

**public class practiceMidtermProblem1 {**

**public static final int SIZE = 5;**

**public static void main(String[] args) {**

**for(int line = 1; line <= SIZE; line++) {**

**for(int i = 1; i <= SIZE - line; i++) {**

**System.out.print(i);**

**}**

**System.out.print("-");**

**for(int j = SIZE - line; j >= 1; j--) {**

**System.out.print(j);**

**}**

**System.out.println();**

**}**

**}**

**}**

**Output:**

**1234-4321**

**123-321**

**12-21**

**1-1**

**-**

**2. What is the output of the following program?**

**public class MixedNumberSalad {**

**public static void main(String[] args) {**

**int x = 4;**

**int eight = 2;**

**int one = 8;**

**String y = "three";**

**String z = "7";**

**String ten = "one";**

**addFour(x, one);**

**subtractFive(ten, y, z);**

**String four = getFive(z, x);**

**System.out.print(four);**

**}**

**public static void addFour(int eight, int x) {**

**System.out.println(x - eight + 10);**

**}**

**public static void subtractFive(String eight, String four, String one) {**

**System.out.print(four + " " + eight + " " + one + " ");**

**}**

**public static String getFive(String x, int z) {**

**z = z + 10;**

**return "x + z";**

**}**

**}**

**Output:**

**14**

**three one 7 x + z 3. Write a program to produce the following pattern. Use a public final static int SIZE; to determine the size of the pattern that needs to be generated. Below are the outputs for sizes 4 and 5:**

**Size 4 Size 5**

**////////////\*\*\*\*\*\*\*\*\\\\\\\\\\\\ ////////////////\*\*\*\*\*\*\*\*\\\\\\\\\\\\\\\\**

**////////\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\\\\\\\\ ////////////\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\\\\\\\\\\\\**

**////\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\\\\ ////////\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\\\\\\\\**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ////\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\\\\**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Write out the whole program code, including the class name and main method.**

**public class SlashesPattern {**

**// The constant size that can be modified to produce different sized patterns**

**public static final int SIZE = 4;**

**public static void main(String[] args) {**

**// Ask yourself: What is changing? What's the pattern?**

**//**

**// We want to print a certain amount of slashes, stars, and backslashes**

**// based on the line number, according on the following equations:**

**// /: ((SIZE\*4) - (line\*4))**

**// \*: (line\*8)**

**// \: ((SIZE\*4) - (line\*4))**

**//**

**// For the slashes, the SIZE\*4 represents the initial total for that given size,**

**// while the line\*4 represents the number of slashes removed, based on the current**

**// line number. Here is a table for reference:**

**// \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**// | SIZE | LINE | / | (SIZE\*4) - (line\*4) |**

**// | 4 | 1 | 12 | (16) - (4) = 12 |**

**// | 4 | 2 | 8 | (16) - (8) = 8 |**

**// | 4 | 3 | 4 | (16) - (12) = 4 |**

**// | 4 | 4 | 0 | (16) - (16) = 0 |**

**// +---------------------------------------------------+**

**// Create SIZE lines**

**// For each of the lines**

**for(int line = 1; line <= SIZE; line++) {**

**// Print ((SIZE\*4) - (line\*4)) many slashes, "/"**

**for(int slash = 1; slash <= ((SIZE\*4) - (line\*4)); slash++) {**

**System.out.print("/");**

**}**

**// Print (line\*8) many stars, "\*"**

**for(int star = 1; star <= (line\*8); star++) {**

**System.out.print("\*");**

**}**

**// Print ((SIZE\*4) - (line\*4)) many backslashes, "\"**

**for(int backslash = 1; backslash <= ((SIZE\*4) - (line\*4)); backslash++) {**

**System.out.print("\\");**

**}**

**// Go to the next line**

**System.out.println();**

**}**

**}**

**}**