

ComS 207: Programming I
Midterm 1, Tue. Feb 7, 2006

Student Name:

Student ID Number:

Recitation Section:

1. True/False Questions (10 x 1p each = 10p)

- (a) This is a valid identifier in Java: 207Rocks TRUE / FALSE
- (b) This is a valid identifier in Java: C\$_207_Rocks TRUE / FALSE
- (c) A constructor method must have the same name as its class. TRUE / FALSE
- (d) A class can have only one constructor method. TRUE / FALSE
- (e) Unicode characters are a subset of the ASCII characters. TRUE / FALSE
- (f) A constructor must be declared for each class. TRUE / FALSE
- (g) A constructor must have a void return type. TRUE / FALSE
- (h) Each object has a state, defined by its variables,
and a set of behaviors defined by its methods. TRUE / FALSE
- (i) Integers can store a maximum of three values at the same time. TRUE / FALSE
- (j) No more than one object can be created from each class. TRUE / FALSE

2. Short Answer Questions (5 x 2p each = 10p)

(a) What is an enumerated type?

(b) What is an object?

(c) What is the difference between a class and an object?

(d) What is autoboxing?

(e) What is the difference between NumberFormat and DecimalFormat?

3. Random Numbers (5 x 2p for each sub-problem = 10p)

(a) Write a snippet of code that produces a random number in the specified range. You can assume that the following line of code occurs somewhere before your snippet of code: `Random rand = new Random();`

Hint: Use the version of the `nextInt` method in the `Random` class which accepts a single integer parameter.

- i. 25 to 49

- ii. -5 to 6

- iii. Generate a random odd integer from 1 to 100

(b) This is the same as (a) but with floating point numbers. The upper boundary is not inclusive in this case however.

- i. 1.5 to 1.7

- ii. -1.2 to -0.2

4. Bug Chase (5p + 10p = 15p)

Find all bugs in the following programs. Circle the location of each bug and write with words what is wrong with the code at that location.

(a) (5p)

```
===== File: Buggy-1.java =====
public class Buggy-1
{
    private static void main (String args) {
        String s = new Sting("John");
        system.out.println("My name is ", s);
        return (s);
    }
}
```

(b) (10p)

```
===== File: Account.java =====
public class Account
{
    private accName;

    Account(String name) {
        accName == name;
    }

    private printInfo() {
        system.out.println("Account Name Is: ", accNaem);
    }
}
```

```
===== File: Bank.java =====
public class myBank
{
    private static void main (String s) {
        account Account = new Account("John Smith");
        account.printInfo();
        return ();
    }
}
```

5. Expressions and Assignments (10 x 2p each = 20p)

For each of the following, write down the value that will be stored in result

```
double result;  
int num1 = 5, num2 = 12, num3 = 2;  
double val1 = 5.0, val2 = 12.0, val3 = 2.0;
```

- (a) `result = num2 / num1;`

- (b) `result = val3 + num2 / num1;`

- (c) `result = num3 + val2 / num1;`

- (d) `result = num2 / num3 / num1;`

- (e) `result = val2 * num1 + num2 / num3;`

- (f) `result = (int) (val2 * (num1 + num2) / num3);`

- (g) `result = (val2 * (num1 + num2)) / num3;`

- (h) `result = val2 * ((num1 + num2) / num3);`

- (i) `result = num1 * num3 * 4 % num2 / val3;`

- (j) `result = num1+++ ++num2;`

6. Programming Projects (TOTAL 65p, but each has a different weight)

(a) Rectangles (15p)

Write a complete Java program which asks the user to enter the coordinates of the lower left and the upper right corners of a rectangle. In other words, the program must read two pairs of floating point numbers which represent the x and y coordinates of two points: $P_{LL} = (x_1, y_1)$ and $P_{UR} = (x_2, y_2)$. The program must then calculate and print on the screen the following four values associated with the rectangle: 1) the length of the horizontal side; 2) the length of the vertical side; 3) the perimeter; 4) the area.

(b) Password Generator (15p)

Write a complete Java program that generates a random password for a computer account. The password must have a length of exactly 5 characters. The first and the fourth characters must be capital letters from the English alphabet (A-Z). The second and the fifth characters must be numbers (0-9). The third character must be a dash (i.e., '-'). Here are two examples of valid passwords: "R2-D2" and "C3-P0".

Hint: The only difficult part in this program should be figuring out how to print a single character (without printing a new line). To do this you can use the following snippet of code where num is defined as an int:

```
System.out.printf("%c", 'A' + num);
```

For example, if num=3 the result should be D; if num=0 the result should be A.

(c) Radians to Degrees (15p)

Write a Java program that converts radians to degrees. The program must ask the user to enter a floating-point number (representing the angle in radians). The program must then convert the angle from radians to degrees and output the result. The output must be formatted into degrees, minutes, and seconds (all of these must be printed as integers, i.e., no decimal points).

Sample input: 2.0

Sample output: 114 degrees, 35 minutes, 29 seconds

Hint 1: $\text{degrees} = (\text{radians} / \text{Math.PI}) * 180.0;$

Hint 2: $114^\circ 35' 29'' = 114 + 35 * (1/60) + 29 * (1/60) * (1/60) = 114.5913889^\circ$

(d) Formatting Names (20p)

Write a Java program that asks the user to enter his three names on a single line. You can assume that the user will always enter his/her names in the following format: “First Middle Last”. In other words, the first character typed in will be the first letter of the first name; the names will be separated with a single space; the last character typed in will be the last character of the last name; and finally all names will have only their first letter capitalized. The program must then format and print the name in the following format: “LAST, First M.” In other words, the last name must be printed first with all capital letters, followed by a comma, followed by the full first name as it was typed in, followed by the first letter of the middle name, followed by a period.

Hint: Use the methods of the String class to find the positions of the separators in the original string. Then chop that string into three separate pieces and print them in the desired order and format.

That's it. Good Luck!

Question	Max	Score
True/False	10	
Short Answer	10	
Random Numbers	10	
Bug Chase	15	
Expressions	20	
Rectangles	15	
Password Generator	15	
Radians to Degrees	15	
Formatting Names	20	
TOTAL:	130	