CompSci 101 S1 2004 – City and Tamaki

Terms Test Model Answers

Question 1 (20 marks)

a) What is printed by the following?

```
System.out.println("5" + 5 + 9);
```

559

(1 mark)

b) What is printed by the following?

```
System.out.println(6 + 3 + "8");
```

98

(1 mark)

c) What is printed by the following?

```
System.out.println((int) 5.3);
```

5

(1 mark)

d) What is printed by the following?

```
System.out.println("\"n\\\n\"");
```

"n
"

(2 marks)

e) What is printed by the following?

```
System.out.println(5 + 9 % 2 * (10 / 4) - 6);
```

1

(2 marks)

f) What is printed by the following?

```
System.out.println(5 / 2.0 * 3 / 5);
```

1.5

(1 mark)

g) What is printed by the following?

```
String string01 = "food looks good";
System.out.println(string01.indexOf("oo",4));
```

6

(2 marks)

h) What is printed by the following?

```
System.out.println(Math.pow(Math.max(4,3), 2));
```

16.0

(2 marks)

i) What is printed by the following?

```
System.out.print((int)3.14159 + "\nis the");
System.out.print("value of pi.");
```

3
is the value of pi.

(2 marks)

j) What is printed by the following?

```
String string02 = "abcdefghij";
System.out.println(string02.substring(3,string02.length()-3));
```

defg

(2 marks)

k) What is printed by the following?

```
String string03 = "  11 22 33  ";
String string04 = string03.trim();
string04+=3;
System.out.println(string04);
```

11 22 333

(2 marks)

l) What is printed by the following?

```
double number01 = 12;
int number02 = 5;
number01 -= number02;
System.out.println(number01);
```

7.0

(2 marks)

Question 2 (10 marks)

Consider the following section of Java source code, and answer the questions below:

```
int x = 5;
int y;
char c = 'a';

y = (int)Math.pow(x,2);
Rectangle aRect = new Rectangle(x,y,30,20);
String rectString = "Rect1" + " is " + aRect.toString();
```

a) List all the *identifiers* which appear in the above code fragment

String, Math, Rectangle,
x, y, c, aRect, rectString
pow, toString

(5 marks)

b) Identify all the variables in the above code fragment and categorize them according to whether
they are primitive or object types.

PRIMITIVE VARIABLES:

x, y, c

OBJECT VARIABLES:

aRect, rectString

(5 marks)

Question 3 (10 marks)

You need to find and correct five errors in the program `Q3`. There is no more than one error on each line of code.

The application program `Q3` is supposed to generate a random number between 0 and 26, and then print out that number of letters from the alphabet. Three examples of the application being executed are shown below – the output must be *identical* to that shown:

Example 1:

```
C:\>java Q3
The first 5 characters of the alphabet are abcde
```

Example 2:

```
C:\>java Q3
The first 0 characters of the alphabet are
```

Example 3:

```
C:\>java Q3
The first 18 characters of the alphabet are abcdefghijklmnopqr
```

The source code for the application `Q3` is given below. Five of the lines of code contain an error of some sort. For each error, you need to clearly circle the error and provide a correction so that the program will compile and execute correctly. You do not need to write out the whole line of source code again, as long as you indicate your corrections clearly.

```
public class Q3 {

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

        String alphabet = "abcdefghijklmnopqrstuvwxyz";

2       int index;

3       index = (int)( Math.random() * 27 );

4       System.out.println("The first " + index + " characters of
                                                    the alphabet");

5       System.out.println(" are " + alphabet.substring(0,
                                                    index));

   }

}
```

(10 marks)

Question 4 (15 marks)

Complete the application `Q4` given below. Your application should perform the following tasks.

1. Prompt the user to enter two integer values as input.
2. Print the word "`Larger:`" followed by the larger of the two input integer values.
3. Print the word "`Difference:`" followed by the difference between the larger and smaller input values.
4. Print the word "`Quotient:`" followed by the *real* value obtained by dividing the first input value by the second input value.

Two examples of the output that must be produced by your `Q4` program are given below. Make sure the output of your application is identical to the format of this output. Values entered by the user are given in bold:

Example 1:

```
C:\>java Q4
Please input first number: 5
Please input second number: 8
Larger: 8
Difference: 3
Quotient: 0.625
```

Example 2:

```
C:\>java Q4
Please input first number: 10
Please input second number: -3
Larger: 10
Difference: 13
Quotient: -3.333333333333333335
```

Complete the source code for this application below. Some variables have been declared for you:

```
import java.io.*;

public class Q4 {

     public static void main(String[] args){

          int number1;
          int number2;
          int difference;
          double quotient;

          System.out.print("Please input first number: ");
          number1 = Integer.parseInt(readInput());

          System.out.print("Please input second number: ");
          number2 = Integer.parseInt(readInput());

          difference = Math.max(number1,number2) –
                                        Math.min(number1,number2);

          quotient = (double)number1/(double)number2;


          System.out.println("Larger: " + Math.max(number1,
                                                        number2));
          System.out.println("Difference:" + difference);

          System.out.println("Quotient:" + quotient);









     }

     private static String readInput() {
          try {
               BufferedReader in = new BufferedReader(
                    new InputStreamReader(System.in));
               return in.readLine();
          }
          catch (IOException e) {}
          return "";
     }
 }
```

(15 marks)

Question 5 (15 marks)

Complete the implementation of the `FruitProfile` class for building simple fruit profiles in a grocery store. The application `Q5` makes use of the `FruitProfile` class as shown below.

```
public class Q5 {

     public static void main(String[] args) {

          FruitProfile fruit1 = new FruitProfile("apple", 10);
          FruitProfile fruit2 = new FruitProfile("pear", 16);

          System.out.println(fruit1.toString());
          fruit1.add(12);
          System.out.println("After adding");
          System.out.println(fruit1.toString());

          System.out.println();
          System.out.println(fruit2.toString());
          fruit2.subtract(5);
          System.out.println("After subtracting");
          System.out.println(fruit2.toString());

          System.out.println("There are currently " +
                         fruit1.getAmount() + " " + fruit1.getName() +
                                                        "(s) in store." );

     }

}
```

The application `Q5` shown above uses the `FruitProfile` class to create two `FruitProfile` objects. When a `FruitProfile` object is constructed, the name of the fruit (a `String`) and the amount in store (an `int`) are both specified. Given a correct implementation of the `FruitProfile` class, the output from the application above should be *exactly* as shown below.

```
C:\>java Q5
Fruit name: apple
Amount: 10
After adding
Fruit name: apple
Amount: 22

Fruit name: pear
Amount: 16
After subtracting
Fruit name: pear
Amount: 11
There are currently 22 apple(s) in store.
```

Complete the implementation of the `FruitProfile` class from the skeleton file given below.

```
public class FruitProfile {

     // instance variables
     private String name;
     private int amount;

     //constructor
     public FruitProfile(String inputName, int inputAmount){
          name = inputName;
          amount = inputAmount;
     }

     // add(): adds input amount to current amount
     public void add(int inputAmount) {
          amount += inputAmount;
     }

     // subtract(): subtracts input amount from current amount
     public void subtract(int inputAmount) {
          amount -= inputAmount;
     }

     // getName : returns name of fruit
     public String getName() {
          return name;
     }

     // getAmount : returns amount of fruit
     public int getAmount() {
          return amount;
     }

     // toString : returns String representation of fruit and
                                                  current amount
     public String toString() {
          return "Fruit name: " + name + "\n" + "Amount: " +
                                                        amount;
     }

}
```

(15 marks)

Question 6 (10 marks)

(a) Evaluate the following `boolean` expressions

(i) `(2 > 3) || (4 == 4) && !((4>5) || (2<3)) || (9!=2)`

true

(1 mark)

(ii) `3==4 && 3 != 4 || !(3 != 3) && !(4 != 4) || 5 <= 4`

true

(1 mark)

(iii) `! ( ! (4 != 6) && !(5 >4))`

true

(1 mark)

(b) DeMorgan's Law consists of the following two equivalences:

```
!(A && B) == !A || !B
!(A || B) == !A && !B
```

Use DeMorgan's Law to help simplify the following expression. The simplified expression should not contain either of the symbols `!` or `!=`

```
!(!(x > 42) && !(x <13))
```

x>42 || x<13

(3 marks)

(c) Complete the truth table for the following `boolean` expression

```
P && Q || !(P || Q) && P
```

P	Q	P&&Q	P||Q	!(P||Q)	!(P||Q)&&P	P&&Q || !(P||Q)&&P
T	T	T	T	F	F	T
T	F	F	T	F	F	F
F	T	F	T	F	F	F
F	F	F	F	T	F	F

(4 marks)

Question 7 (10 marks)

(a) Translate the following sentences from English to Java

(i) If `a` is less than 7 or `b` is greater than 7 then print out "`greater than 7`"

```
if( a<7 || b>7)
        System.out.println("greater than 7");
```

(1 mark)

(ii) Print "`too easy`" unless `finalMark` is less than 50

```
if( !(finalMark < 50) )
        System.out.println("too easy");
```

(1 mark)

(iii) Print "`too hard`", but only if `finalMark` is less than 50

```
if( finalMark < 50 )
        System.out.println("too hard");
```

(1 mark)

(iv) If either `a` or `b` is greater than 8 and if neither `c` nor `d` is less than 8, then print "`abcd`"

```
if( (a>8 || b > 8) && !(c<8 || d<8) )
        System.out.println("abcd");
```

(2 marks)

(b) What is the output of the following code?

```
int x = 4;
int y = 5;
int z = 6;
if(x>y || y>z)
x = 0;
else
if(x<y && y<z)
if(x==y)
z=y;
else
y = 0;
else
if(y==0 || x<z)
z = 0;
System.out.println(x + " " + y + " " + z);
```

```
4 0 6
```

(5 marks)

Question 8 (5 marks)

Assume that variables `r` and `s` have been declared as follows:

```
Rectangle r = new Rectangle(10,20,30,40);
Rectangle s = new Rectangle(10,20,30,40);
```

(i) Write an if statement which will print out "`equal value`" if `r` is equal *in value* to `s`.

```
if ( r.equals(s) )
        System.out.println("equal value");
```

(2 marks)

(ii) Write an if statement which prints out "`equal reference`" if `r` is equal *in reference* to `s`.

```
if ( r==s )
        System.out.println("equal reference");
```

(2 marks)

(iii) Are the two variables `r` and `s` equal in reference, equal in value, or both?

```
equal in value
```

(1 mark)

Question 9 (5 marks)

What is the output of the following code?

```
int x = 25;
int y = 10;
int count = x;

System.out.println("Before: " + count);

while(count > y){
     if(count%5==0)
          System.out.println(count);
     count--;
}

System.out.println("After: " + count);
```

Before: 25
25
20
15
After: 10

(5 marks)