First Name (Print): _____ **Last Name (Print):** _____

Student Number: _____

The Irving K. Barber School of Arts and Sciences
COSC 111 Midterm 1 Winter Term II 2016

Instructor: Dr. Bowen Hui

Monday, February 15, 2016

Time: 5:15pm - 6:30pm

Instructions:

This is a closed book exam. No calculators are allowed. Turn off your cell phone. Have your student ID card out on your desk.

This exam has 9 pages (including this cover page).

Question	Score	Possible Marks
Part 1		10
Part 2 Question 1		8
Part 2 Question 2		5
Part 2 Question 3		8
Part 3		9
Total		40

Part 1: Multiple Choice (10 points)

Circle one answer for each question below.

1. What is the return value of `"SELECT".substring(4, 4)`?
   (a) an empty string
   (b) C
   (c) T
   (d) E

2. The expression `4 + 20 / (3 - 1) * 2` is evaluated to
   (a) 4
   (b) 20
   (c) 24
   (d) 9
   (e) 25

3. According to Java naming convention, which of the following names can be variables?
   (a) `FindArea`
   (b) `findArea`
   (c) `totalLENGTH`
   (d) `TOTAL_LENGTH`
   (e) `class`

4. Which of the following is equivalent to the boolean expression `x != y`?
   (a) `!( x == y )`
   (b) `x > y && x < y`
   (c) `x > y ^ x < y`
   (d) `x >= y || x <= y`

5. To declare a constant `MAX_LENGTH` inside a method with value 99.98, you write:
   (a) `final MAX_LENGTH = 99.98;`
   (b) `final long MAX_LENGTH = 99.98;`
   (c) `double MAX_LENGTH = 99.98;`
   (d) `final double MAX_LENGTH = 99.98;`

6. Note that the ASCII value for character A is 65. What does the expression `"A" + 1` evaluate to?

   (a) 66
   (b) B
   (c) A1
   (d) Syntax error (illegal expression)

7. Suppose x = 1, y = 1, and z = -1. What is the printout of the following statement? Hint: Indent the statement correctly first.

```
String fruit = "unassigned";
if( x > 0 )
{
   fruit = "apple";
   if( y > 0 )
      System.out.print( " " + fruit );
else if( z > 0 )
       fruit = "orange";
       System.out.print( " " + fruit );
}
else
  fruit = "banana";
  System.out.println( " " + fruit );
```

   (a) `apple apple`
   (b) `apple apple apple`
   (c) `apple orange`
   (d) `apple apple orange`
   (e) `apple banana`

8. Assume x = 4, Which of the following is true?

   (a) `!(x == 4)`
   (b) `x != 4`
   (c) `x == 5`
   (d) `x != 5`

9. Which of the following code displays the area of a circle if the radius is positive?

   (a) `if (radius != 0) System.out.println(radius * radius * 3.14159);`
   (b) `if (radius >= 0) System.out.println(radius * radius * 3.14159);`
   (c) `if (radius > 0) System.out.println(radius * radius * 3.14159);`
   (d) `if (radius <= 0) System.out.println(radius * radius * 3.14159);`

10. Which of the following expression results in 45.37?

    (a) `( int )( 45.378 * 100 ) / 100`
    (b) `( int )( 45.378 * 100 ) / 100.0`
    (c) `( int )( 45.378 * 100 / 100 )`
    (d) `( int )( 45.378 ) * 100 / 100.0`

Part 2: Short Answers (21 points)

Question 1. (8 points)

Consider the following code stored inside HelloWorld.java:

```
public class HelloWorld
{
  public static void main( String args[] )
  {
    int x = 5;
    int y = 2;
    System.out.println( "sum is " + x * y );

    int sq = x * x;
    System.out.println( "x = " + x + " and sq = " + sq );

    String str = "averylongwordwithnospace";
    System.out.println( str.charAt( 3 ) + str.substring( 2, 5 ) );

    System.out.println( 45 / 4 );

    double a = 10.1;
    int    b = ( int )a;
    System.out.println( "a is " + a + " and b is " + ( double )b );

    boolean found = false;
    if( found = true )
      System.out.println( "Lost item has been found!" );
    else
      System.out.println( "Item is still lost!" );

    x = 14;
    y = 15;
    System.out.println( y % 2 == 0 || x % 2 == 0 );

    System.out.println( "peter".toUpperCase().equals( "Peter" ) );
  }
}
```

When this program is run, what is displayed on the screen?

Grading scheme:

- **[1 pt each]** Correct output for each print statement

Question 2. (5 points)

Convert the following mathematical expression into an equivalent Java statement:

$$x = \frac{QI - \sqrt{a^2 - \frac{3}{71}(b + 4c)}}{2b} \tag{1}$$

You can assume that the variables `x`, `a`, `b`, and `c` are of type `double` and `QI` is a constant of type `int`. Recall that the `Math` class has a square root method named `sqrt` which takes an argument of type `double` and returns the computation as a `double`.

Grading scheme:

- **[3 pts]** translation of numerator
- **[1 pt]** translation of denominator
- **[1 pt]** overall syntax

Question 3. [8 points]

Given three integer variables named `num1`, `num2`, and `num3`, write the code that determines which variable stores the smallest number. You don't know what numbers are actually stored in these variables, but you can assume that all the numbers are different from each other. When you find the smallest number, store the result into another integer variable named `min`.

Grading scheme:

- **[1 pt]** Code works for the case: $num1 < num2 < num3$
- **[1 pt]** Code works for the case: $num1 < num3 < num2$
- **[1 pt]** Code works for the case: $num2 < num1 < num3$
- **[1 pt]** Code works for the case: $num2 < num3 < num1$
- **[1 pt]** Code works for the case: $num3 < num1 < num2$
- **[1 pt]** Code works for the case: $num3 < num2 < num1$
- **[1 pt]** correct use of boolean expressions
- **[1 pt]** overall syntax

Part 3 Long Answer [9 points]

A "magic 8 ball" is a ball that lets you ask it a life question, shake it, and see the answer to your question appear on the ball. Here, we will write a simple version that behaves as follows:

- Ask the user to enter a life question
- If the first word of that question starts with the letter `W` or `w`, then your program will have 50% chance returning the output "Outlook not so good" and 50% chance returning "Signs point to yes"
- Otherwise, if there question has fewer than 50 letters (including spaces), then your program will return the output "Don't count on it"
- Otherwise, your program will output "You should reflect on it first"

To get full marks, you need to write a `complete` Java program that runs without errors if we were to type it in exactly as you have written in. Include any libraries that the program needs to import in order for it to work.

Grading scheme:

- **[1 pt]** importing necessary libraries
- **[1 pt]** class and main method properly defined
- **[1 pt]** reading input question from user
- **[1 pt]** checking question if it starts with "W" or "w"
- **[1 pt]** checking question if it is shorter than 50 letters
- **[2 pts]** randomly generating two possible outputs with 50% each
- **[1 pt]** displaying appropriate output for remaining cases
- **[1 pt]** overall syntax

Common Methods and Definitions

- From the `Math` class:
  - `double random()`
    Returns a `double` value greater than or equal to 0.0 and less than 1.0
  - `int round( float a )`
    Returns the closest `int` value to `a`, with ties rounding up
  - `double pow( double a, double b )`
    Returns the value of `a` raised to the power of `b`
  - `double exp( double a )`
    Returns Euler's number e raised to the power of `a`
  - `double sqrt( double a )`
    Returns the correctly rounded positive square root of `a`
- From the `Character` class:
  - `boolean isDigit( char ch )`
    Returns true if `ch` is a digit, and false otherwise
  - `boolean isLetter( char ch )`
    Returns true if `ch` is a letter, and false otherwise
  - `boolean isLetterOrDigit( char ch )`
    Returns true if `ch` is a letter or a digit, and false otherwise
  - `boolean isLowerCase( char ch )`
    Returns true if `ch` is a lowercase letter, and false otherwise
  - `boolean isUpperCase( char ch )`
    Returns true if `ch` is an uppercase letter, and false otherwise
- From the `String` class:
  - `char charAt( int index )`
    Returns the character at position `index` of the string
  - `int indexOf( int ch )`
    Returns the index position within the string of the first occurrence of `ch`
  - `int lastIndexOf( int ch )`
    Returns the index position within the string of the last occurrence of `ch`
  - `int length()`
    Returns the number of characters in the string
  - `boolean startsWith( String prefix )`
    Returns true if the string starts with `prefix`, and false otherwise
  - `boolean endsWith( String suffix )`
    Returns true if the string ends with `suffix`, and false otherwise
  - `int compareTo( String str )`
    Returns an integer result for comparing two strings lexicographically
  - `int compareToIgnoreCase( String str )`
    Returns an integer result for comparing two strings lexicographically while ignoring case differences
  - `boolean equals( Object stringObject )`
    Returns true if the string is the same as `stringObject`, and false otherwise
  - `boolean equalsIgnoreCase( String str )`
    Returns true if the string is the same as `str`, and false otherwise
  - `String substring( int beginIndex )`
    Returns a new string that is a substring starting at position `beginIndex` of this string
  - `String substring( int beginIndex, endIndex )`
    Returns a new string that is a substring starting at position `beginIndex` and up to `endIndex` of this string
  - `String toLowerCase()`
    Converts all the characters in this string to lower case

  - `String toUpperCase()`
    Converts all the characters in this string to upper case

- From the `Scanner` class:
  - `int nextInt()`
    Scans the next token of the input as an `int`
  - `double nextDouble()`
    Scans the next token of the input as a `double`
  - `String next()`
    Returns the next complete token as a `String`
  - `String nextLine()`
    Returns everything in the current line as a `String`