INSTRUCTIONS: This is the second homework for CS170 (Section 003) and covers the following material: introduction to Java, variables, expressions, Java library. Read the instructions carefully and fill in your answers and solutions. You need to turn in this homework on paper.

HONOR CODE: Like all work for this class, the Emory Honor Code applies. You should do your own work on all problems, unless you are explicitly instructed otherwise. If you get stuck or have questions, ask your instructor or a TA for help.

I commit to uphold the ideals of honor and integrity by refusing to betray the trust bestowed upon me as a member of the Emory community. I have also read and understand the requirements and policies outlined above.

Signature: ____________________________

Problem 1

1. (10 points) Topic: Java Expressions

Fill out the following table. Evaluate the Java expression in the first column and put the result in the second column. Assume that each expression is evaluated independently (ie, not in sequence). In the third column indicate the datatype of the result. The first row has been done for you.

\[ \text{Note: It is recommended that you do this question without Java compiler. You can answer using the Java compiler, but if you do this, you will miss out on a chance to learn. You will be asked to do similar problems on the midterm where you won’t have access to a Java compiler. I do recommend you write these statements inside a Java program after you have done the homework. You can check your answers — if you have errors, understand why.} \]

\begin{verbatim}
double a = 4.5, b = 29.0;
int c = 35, d = 6;
\end{verbatim}
### Problem 2

2. (20 points) Topic: Java expressions

Write Java expressions to evaluate the following formulas (assume that all the variable used in the formula were declared and initialized before).

<table>
<thead>
<tr>
<th>Java expression</th>
<th>Result</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c + d</td>
<td>41</td>
<td>int</td>
</tr>
<tr>
<td>c - d * d</td>
<td>-1</td>
<td>int</td>
</tr>
<tr>
<td>c / d</td>
<td>5</td>
<td>int</td>
</tr>
<tr>
<td>1.0 * c / d</td>
<td>5.833333333333333</td>
<td>double</td>
</tr>
<tr>
<td>c / d * 1.0</td>
<td>5.0</td>
<td>double</td>
</tr>
<tr>
<td>b / d</td>
<td>4.833333333333333</td>
<td>double</td>
</tr>
<tr>
<td>a++ - –d</td>
<td>10.5</td>
<td>double</td>
</tr>
<tr>
<td>b - a &lt;= c - d</td>
<td>true</td>
<td>bool</td>
</tr>
</tbody>
</table>

### Problem 3

3. (20 points) Topic: Java library
Access the Java API documentation from the class website. It is linked in the left hand sidebar. In the Java docs, look in the top left corner of the page for a list of packages. Write a brief description for each package listed below. Also choose one of the classes, implemented in this package and describe it in a sentence (what it is needed for, if you don’t understand what the documentation says, just write the first sentence of the description).

**Example:** java.awt.color: Provides classes for color spaces. Class ColorSpace is used to serve as a color space tag to identify the specific color space of a Color object.

- **java.lang**

  **Solution:** Provides classes that are fundamental to the design of the Java programming language.
  java.lang.String: The String class represents character strings. All string literals in Java programs, such as “abc”, are implemented as instances of this class.

- **java.io**

  **Solution:** Provides for system input and output through data streams, serialization and the file system.

- **java.util**

  **Solution:** Contains the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (a string tokenizer, a random-number generator, and a bit array).
  java.util.Scanner: A simple text scanner which can parse primitive types and strings using regular expressions.

- **java.math**

  **Solution:** Provides classes for performing arbitrary-precision integer arithmetic (BigInteger) and arbitrary-precision decimal arithmetic (BigDecimal).
  java.math.BigInteger class represents immutable arbitrary-precision integers. BigInteger provides analogues to all of Java’s primitive integer operators, and all relevant methods from java.lang.Math. Additionally, BigInteger provides
Problem 4

4. (25 points) Topic: Java programming

Write a Java program, that reads in 6 double typed numbers $a_1, b_1, c_1, a_2, b_2, c_2$ and prints the solution to the system of 2 linear equations:

$$a_1 x + b_1 y = c_1$$
$$a_2 x + b_2 y = c_2$$

Note: Please refer to this Wikipedia page for more information on how to solve such a system.

Name your program Question4.

```java
import java.util.Scanner;

public class Question4 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        // Declare variables for equations coefficients.
        double a1, a2, b1, b2, c1, c2;
        System.out.print("Please enter a1 b1 c1 of the first equation: ");
        a1 = input.nextDouble();
        b1 = input.nextDouble();
        c1 = input.nextDouble();
        System.out.print("Please enter a2 b2 c2 of the first equation: ");
        a2 = input.nextDouble();
        b2 = input.nextDouble();
        c2 = input.nextDouble();

        // Well, we are actually ignoring no solutions cases.
        // We need to work out expressions for x and y.
        // From the first equation $y = (c1 - a1*x) / b1$
        // Substitute this to the second equation
        // and get an expression for x:
        double x = (c2 - c1 * b2 / b1) / (a2 - a1 * b2 / b1);

        // And now we can calculate y
        double y = (c1 - a1 * x) / b1;

        System.out.println("Solution is x = " + x + " y = " + y);
    }
}
```
Problem 5

5. (25 points) Topic: Java programming and Java library

Write a Java program, that reads 3 float values (a, b and c) and prints their minimum and maximum. Name your program Question5.

Hint: Look up the java.lang.Math class in the Java documentation. Among the methods of this class you may find something that can help you calculate minimum and maximum of 2 numbers. How you can use this to calculate the minimum and maximum of 3 numbers?

```java
import java.util.Scanner;

public class Question5 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int a, b, c;
        System.out.print("Please enter numbers a b c: ");
        a = input.nextInt();
        b = input.nextInt();
        c = input.nextInt();

        // We will use Math.min and Math.max functions.
        // They take 2 parameters, so we need to call it
        // twice to get the result.
        int min = Math.min(a, Math.min(b, c));
        int max = Math.max(a, Math.max(b, c));

        System.out.println("Maximum value is "+max+", minimum is "+min);
    }
}
```