• **INSTRUCTIONS:**

  – Keep your eyes on your own paper and do your best to prevent anyone else from seeing your work.
  – Do NOT communicate with anyone other than the professor/proctor for ANY reason in ANY language in ANY manner.
  – This exam is closed notes, closed books, and no calculator.
  – Turn all mobile devices off and put them away now. You cannot have them on your desk.
  – You cannot use the Internet!
  – Write neatly and clearly indicate your answers. What I cannot read, I will assume to be incorrect.
  – Stop writing when told to do so at the end of the exam. I will take 5 points off your exam if I have to tell you multiple times.
  – Academic misconduct will not be tolerated. Suspected academic misconduct will be immediately referred to the Emory Honor Council. Penalties for misconduct will be a zero on this exam, an F grade in the course, and/or other disciplinary action that may be applied by the Emory Honor Council.

• **TIME:** This exam has just 1 problem. You will have 15 minutes to complete this exam.

---

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: ___________________________

<table>
<thead>
<tr>
<th>Question:</th>
<th>1</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>25</td>
<td>25</td>
</tr>
</tbody>
</table>

Score: |
1. (25 points) Imagine, that you work for Facebook and you are to write a program that will estimate emotional state of some user. The only information you have about the user is the latest message he posted (a message is just some string). You know that people use smileys (for example, :-) or :-( ) a lot in their communications and you have the following idea: you want to count how many open and closed parentheses does the message contain and then if there are more open parentheses ‘(‘, then the person is sad and if the message contains more closed parentheses ’)’, then the user is happy. But unfortunately, not all messages have enough parentheses to make a conclusion which will be trustworthy. If the message has less than 5 parentheses (total), you cannot make a good conclusion on user’s mood.

So, to implement this idea you need to write a Java program, that asks user to enter a message (a message is just a string, one line of text that can contain spaces). Then your program should count how many open ‘(‘ and closed ‘)’ parentheses does the message contain. If the total number of parentheses (open and closed together) is less than 5, then you need to print back the message ”User’s mood cannot be determined.”, otherwise if the message contains more open parentheses ‘(‘ you need to print back the message ”The user is sad.”, if there are more closed parentheses ’)’ you need to print a message ”The user is happy.”, and finally if you have the same number of open and closed parentheses you need to print ”The user is calm.”

Examples:

```java
>>> java DetermineMood
Enter a message: Today I passed my exam ))))))
The user is happy

>>> java DetermineMood
Enter a message: Hi ) I don’t know how to solve this problem ((((
The user is sad.

>>> java DetermineMood
Enter a message: Hooray ))))
User’s mood cannot be determined.

>>> java DetermineMood
Enter a message: I like and hate Java at the same time (-.-) (-.-) (-.-)
The user is calm.
```
Solution:

```java
import java.util.Scanner;
public class DetermineMood {
    public static void main(String[] args) {
        // Declare a variable for scanner and create a new Scanner
        Scanner in = new Scanner(System.in);
        // Print a message to the user and read the line of input
        System.out.print("Enter a message: ");
        String message = in.nextLine();

        // Variables, where we will store the number of open parentheses
        int openPar = 0;
        // and number of closed parentheses. Initialized to 0
        int closedPar = 0;

        // To check all characters in a string,
        // we should go over all characters
        // We can do this using a loop,
        // which runs message.length() number of times
        for (int i = 0; i < message.length(); ++i) {
            // For each character we check
            // if it is open or closed parentheses
            if (message.charAt(i) == '(')
                // and increment the corresponding variable
                openPar++;
            else if (message.charAt(i) == ')')
                closedPar++;
            // for other possible characters
            // we don’t need to do anything
        }

        // Check all the conditions, mentioned in the problem
        // And print the corresponding message
        if (openPar + closedPar < 5)
            System.out.println("User’s mood cannot be determined.");
        else if (openPar < closedPar)
            System.out.println("The user is happy.");
        else if (openPar > closedPar)
            System.out.println("The user is sad.");
        else
            System.out.println("The user is calm.");
    }
}
```