Name (print): ______________________________

• 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.
  – 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 7 questions on 12 pages including the title page. Please check to make sure all pages are included. You will have 50 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>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>12</td>
<td>20</td>
<td>16</td>
<td>12</td>
<td>10</td>
<td>10</td>
<td>20</td>
<td>100</td>
</tr>
<tr>
<td>Score:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. (12 points) There are 3 kinds of programming errors: syntax errors, runtime errors and logical errors. For each of the following pieces of code identify if it contains an error. If it contains an error, describe it and identify the error as a syntax, runtime or logical error.

(a) // This code simply prints Hello World
   System.out.print(Hello World);

   \[ Solution: \text{The code contains a syntax error: missing double quotes around the “Hello World” string.} \]

(b) // This code should print "abc", which is a substring of "abcdef"
   String str = "abcdef";
   System.out.println(str.substring("abc"));

   \[ Solution: \text{This code contains syntax error. There is no such substring method that takes a parameter of type String.} \]

(c) // This program prints each digit of a number on separate lines
    // starting from the most significant digit.
    // Assume Scanner class was imported
    Scanner in = new Scanner(System.in);
    int a = in.nextInt();
    while (a != 0) {
      System.out.println(a % 10);
    }

   \[ Solution: \text{This code contains logical error. Since } a \% 10 \text{ gives the last digit of a number, the digits will be printed starting from the least significant one.} \]

(d) //This code converts a phone number in the format XXX-XX-XXXX to a single
    // integer. e.g. "123-45-6789" => 123456789
    String phone = "123-45-6789";
    int phone_num = Integer.parseInt(phone);

   \[ Solution: \text{This code will cause a runtime error, because a string ”123-45-6789” cannot be converted to an integer as it contains nondigit characters.} \]

2. (20 points) Evaluate each expression. Then give the result of the evaluation and the data type of the result. If the expression cannot be evaluated or is not proper Java syntax, you may simply write “error” for the value. The first row has been done for you.
boolean b = true;
int i1 = 8, i2 = 2;
double d1 = 0.2, d2 = 3;
String s1 = "CS", s2 = "170", s3 = "00", s4 = "3";
char c1 = '7';

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>7/2</td>
<td>3</td>
<td>int</td>
</tr>
<tr>
<td>(i1 &gt;= i2) ? -1L : 1L</td>
<td>-1</td>
<td>long</td>
</tr>
<tr>
<td>i2 % i1</td>
<td>2</td>
<td>int</td>
</tr>
<tr>
<td>(int)s4 + 2</td>
<td>error</td>
<td></td>
</tr>
<tr>
<td>i1/d2</td>
<td>2.6</td>
<td>double</td>
</tr>
<tr>
<td>s2 + s4</td>
<td>&quot;1703&quot;</td>
<td>String</td>
</tr>
<tr>
<td>s2.compareTo(s3) &gt;= 0</td>
<td>true</td>
<td>boolean</td>
</tr>
<tr>
<td>s1.charAt(1)</td>
<td>'S'</td>
<td>char</td>
</tr>
<tr>
<td>Integer.parseInt(s4 + i1) + s3.length()</td>
<td>40</td>
<td>int</td>
</tr>
<tr>
<td>!b</td>
<td>false</td>
<td>boolean</td>
</tr>
<tr>
<td>i1 + i2 + s1</td>
<td>10CS</td>
<td>String</td>
</tr>
<tr>
<td>s1 + i1 + i2</td>
<td>CS82</td>
<td>String</td>
</tr>
<tr>
<td>(s1 + s2 + s3 + s4).substring(1,4)</td>
<td>S17</td>
<td>String</td>
</tr>
<tr>
<td>(int)d2*d1</td>
<td>0.6</td>
<td>double</td>
</tr>
<tr>
<td>Double.ToString(d1)</td>
<td>&quot;0.2&quot;</td>
<td>String</td>
</tr>
<tr>
<td>s2.charAt(1) != c1</td>
<td>false</td>
<td>boolean</td>
</tr>
</tbody>
</table>
3. (16 points) Assume the statements below are part of a Java program that compiles and runs. What is the output if the user types the following: 20 15

```java
Scanner in = new Scanner(System.in);
int num1 = in.nextInt(), num2 = 0, num3 = 0;

if (num1 < 20 || num1 > 10) {
    num2 = in.nextInt();
    System.out.println("num2 gets the value");
} else {
    num3 = in.nextInt();
    System.out.println("num3 gets the value");
}
if (num2 != num3) {
    System.out.println("num2 doesn’t equal num3");
} else {
    System.out.println("num2 equals num3");
}

if (num1 <= 10) {
    System.out.println("aaa");
} else if (num1 <= 20) {
    System.out.println("bbb");
} else if (num1 <= 30) {
    System.out.println("ccc");
} else {
    System.out.println("ddd");
}

switch (num1) {
    case 10:
        System.out.println("AAA");
    case 20:
        System.out.println("BBB");
    case 30:
        System.out.println("CCC");
        break;
    default:
        System.out.println("DDD");
        break;
}

if (num1 == 0 || num2 == 0 || num3 == 0) {
    System.out.println("We have zeros");
    if (num2 % 10 > 10)
        System.out.println("Are you sure?");
    System.out.println("Hi there!");
    if (num3 % 10 == 0)
        System.out.println("I like num3");
} else {
    System.out.println("We don’t have zeros");
}
System.out.print("I’ll get " + (4 * num2 + 2 * num1) + " for the exam.");
```
Solution:

num2 gets the value
num2 doesn’t equal num3
bbb
BBB
CCC
We have zeros
Hi there!
I like num3
I hope to get 100 for this exam.

4. (12 points) You are given several code snippets. In each snippet you have some loop statement. The task is to convert for loops into equivalent while loops (if the snippet contains a for loop), or to convert while loops into equivalent for loops (if the snippet contains a while loop). The output shouldn’t change. One example is completed for you.

(a) // Convert the for loop below to an equivalent while loop.
   for (int i = 0; i < n; i++) {
     System.out.println(i+1);
   }

   Correct solution:
   int i = 0;
   while (i < n) {
     System.out.println(i+1);
     i++;
   }

(b) int i = 1;
   while (i < n) {
     System.out.println(i);
     i+=2;
   }

   Solution:
   for (int i = 1; i < n; i+=2) {
     System.out.println(i);
   }

(c) // in is a Scanner variable
   for (int n = in.nextInt(); n > 0; n /= 10) {
     System.out.println(n % 10);
   }
Solution:

```java
int n = in.nextInt();
while (n > 0) {
    System.out.println(n % 10);
    n /= 10;
}
```

(d) int i = 10;
    while (true) {
        while (i <= 0) break;
        System.out.println(i);
        --i;
    }

Solution:

```java
for (int i = 10; i > 0; --i) {
    System.out.println(i);
}
```
5. (10 points) You are given a snippet of Java program. The task is to draw a flow-chart diagram for this code.

```java
for (int i = 0; i < 10; ++i) {
    if (i % 2 == 0)
        continue;
    System.out.println(i);
    if ((i + 2*i) % 3 == 0)
        break;
}
```

Solution:
6. (10 points) One of your friends decided to make all the decisions in his life by flipping a coin (you should probably talk him out of this idea, but this is a different story). When you met him today in the CS lab, you asked if he wants to have lunch at Panera Bread or Zoe’s Kitchen. Unfortunately, he doesn’t have any coins left and neither do you. But since you are taking CS170 you decided to help your friend and write a program that will flip a coin for him.

So, in this exercise you need to write a program, that prints Heads or Tails randomly with equal probability.

Note: to generate a random number you can use the Math.random() method. This method returns a double typed random number from 0 to 1 (1 is not included). Each time you call Math.random() a different number is generated.

```java
class HeadsTails {
    public static void main(String[] args) {
        double number = Math.random();

        /*---------- Your code here ----------*/
    }
}
```
Solution:

```java
if (number < 0.5) {
    System.out.println("Heads");
} else {
    System.out.println("Tails");
}
```
7. (20 points) Another friend of yours likes strings very much. Once he got a big string for his birthday (yes, by string I mean just a sequence of characters). The string consists only of lowercase latin letters. For example: abbaahhzzz. The thing that amazes your friend are substrings that have the same character repeated several times, e.g. aa, bbb, yyyyyy, etc. Help your friend to find the length of longest substring which consists of multiple repetition of some character.

You need to write a program, that asks user to enter a string of lowercase latin letters (you don’t need to check if the value user enters is correct, you can assume that user always enters a correct string) and prints the length of the longest substring made of the same character repeated multiple times (possibly just 1, see examples for details).

Examples:

```java
>>> java CountSubstring
Enter a string (lowercase characters only): abcd
The length of the longest substring made of the same character = 1

>>> java CountSubstring
Enter a string (lowercase characters only): aabc
The length of the longest substring made of the same character = 2

>>> java CountSubstring
Enter a string (lowercase characters only): abcdddeeffgggh
The length of the longest substring made of the same character = 4

>>> java CountSubstring
Enter a string (lowercase characters only): a
The length of the longest substring made of the same character = 1
```
import java.util.Scanner;

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

        System.out.println("Enter a string (lowercase characters only): ");
        String s = in.next();

        /*------- Your code here -------*/
    }
}

}
Solution:

```java
int best_length = 0;
int current_length = 1;
// Traverse all characters starting from the second one.
for(int i = 1; i < s.length(); ++i) {
    // If the new character
    if (s.charAt(i) != s.charAt(i-1)) {
        best_length = Math.max(best_length, current_length);
        current_length = 1;
    } else {
        current_length++;
    }
}
best_length = Math.max(best_length, current_length);
System.out.println("The length of the longest substring made of the same character = "+ best_length);
```