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 10 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>10</td>
<td>21</td>
<td>10</td>
<td>15</td>
<td>10</td>
<td>14</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>

Score: 1
1. (10 points) Definitions:
Please give basic (brief) definitions for 5 of the 6 following terms. You may include an example if it is helpful to your answer. Please mark the one that you do not want to answer; otherwise, I will assume that the last term is to be discarded.

- Algorithm

- Assignment statement

- Method

- Variable

- Compiler

- Primitive data type
2. (21 points) Evaluate the following expressions. Fill in the value of the evaluation and the data type of the result. The first row has been completed as an example. Assume that expressions are evaluated independently.

```java
int x = -2, y = 4;
double f1 = 0.5, f2 = 6.0;
char letter = 'a';
String str1 = "50", str2 = "Class";
```

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>x + 1</td>
<td>-1</td>
<td>int</td>
</tr>
<tr>
<td>2 + y / 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f1 * 2 + y / 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(x - 2) + str1 + (y + 2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(int) f1 * 2 + (double) x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>str1.length() + 1 == --y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>str2.charAt(1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;&quot; + letter + '1'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x == 3 &amp;&amp; f1 &lt; 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>('d'&gt; letter)</td>
<td></td>
<td>false</td>
</tr>
<tr>
<td>(char) (str2.charAt(0) + 2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>str2.substring(1,3) + 'n'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>str1 + Double.parseDouble(str1)/ 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(f2 == 6.0</td>
<td></td>
<td>x == y) &amp;&amp; letter != 'A'</td>
</tr>
<tr>
<td>f2++ / (--y ) + x++</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3. (10 points) Suppose you are given the following variable definitions:

```java
int intNum = 2;
double doubleNum = 4.0;
```

write the values that will be printed to the console (terminal). If a statement has syntax errors, write "Error" and provide the reason (or a way to correct the error).

a) Code:
```
System.out.println("1 + 1 = " + intNum);
System.out.println("b : " + (intNum * 3));
```

1 + 1 = 2
b : 6

b) Code:
```
double z = "40" - 2;
```

Error.
Subtraction (the - operator) is not defined/supported on an operand of type String.
Correction: Use the method `Double.parseDouble()`. 

c) Code:
```
if (doubleNum == 2) {
    System.out.println(intNum + "\n + 6);
}
System.out.println("" + intNum);
```

2

d) Code:
```
if (intNum > 0) {
    System.out.println("Period");
} else (intNum == 0) {
    System.out.println("Comma");
} else {
    System.out.println("Mark");
}
```

Error.
There cannot be two else statements, and also a condition after else with no keyword if.
Correction: Add a keyword `if` before the first else.

e) Code:
```
while (intNum > 0) {
    if (doubleNum != 2) {
        System.out.print(doubleNum);
    }
    intNum--;
}
```

4.04.0
4. (15 points) Assume the statements below are part of a Java program. What is the output if the user types in 10?

```java
Scanner in = new Scanner(System.in);
int x = in.nextInt();
if (x + 2 > 0 && x - 2 <= 10){
    System.out.println("Grievous");
    if(x * 2 < 30){
        if(x >= 5){
            System.out.println("Omega System");
        }
        System.out.println("Ollivander");
    } else {
        System.out.println("Obi-Wan");
    }
    System.out.println("Darth Vader");
}
System.out.println("Le" + "ia");
x += 2;
switch(x){
case 11:
    System.out.println("Hagrid");
case 12:
    System.out.println("Umbridge");
case 13:
    System.out.println("Chewbacca");
    break;
case 14:
    System.out.println("Yoda");
    break;
default:
    System.out.println("Lupin");
}
System.out.println("Kamino" + "\n" + "Tatooine" + "\nOzzel");
if(10 > x){
    System.out.println("Coruscant");
} else {
    System.out.println("Alderaan");
    switch (x + 3) {
    case 15:
        System.out.println("Luke");
        break;
    case 16:
        System.out.println("C-3PO");
        break;
    }
    if (--x - 1 != 10) {
        System.out.println("Hoth");
    } else {
        System.out.println("Lars");
    }
}
```
5. (10 points) The following code is a Java program that compiles and run. Please provide the output of the program (i.e. what values are printed to the console/terminal).

```java
public class LoopUsage
{
    public static void main(String[] args)
    {
        int a = 1;
        int i = 4;

        while ( i > 0 )
        {
            if ( (i - 1) > 0 ) {
                a += 2;
                System.out.println(i);
            } else {
                System.out.println(i + 2);
            }

            i--;
            System.out.println(i);
        }

        System.out.println(i);
        System.out.println(a);
    }
}
```
6. (14 points) Complete the program below. The program should prompt the user to enter 50 floating-point numbers. The program should output the arithmetic mean (the average) of those numbers.

*Hint:* Use the Scanner method `nextDouble()` and update the running sum in a loop. Then compute the mean and output it to the console/terminal. You might need some additional variables to keep track of the sum and how many numbers the user has entered so far.

```java
import java.util.Scanner;
public class Average50 {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);

        /*---------- Your code here ----------*/
    }
}
```
7. (20 points) Complete the following program. The user enters a String str. If the first character of the string is a digit, the program should output the length of the string. If the first character of the string is a lower-case alphabetical letter, the program should print how many times that character occurs in the string. For other cases, the program should print "Nothing".

For example:

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Output</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>abae23aa1</td>
<td>4</td>
<td>'a' appears 4 times.</td>
</tr>
<tr>
<td>c08xyc</td>
<td>2</td>
<td>'c' appears 2 times.</td>
</tr>
<tr>
<td>1acde</td>
<td>5</td>
<td>'1' is a digit and the string length is 5.</td>
</tr>
<tr>
<td>Abdsd xyts</td>
<td>Nothing</td>
<td>'A' is not a lower-case letter or digit.</td>
</tr>
</tbody>
</table>

(Part of the code has been written for you.)

```java
import java.util.Scanner;
public class StringManipulation {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("Please enter a string: ");
        String str = in.next();
        /*----------- Your code here -------------*/
    }
} // Use the next page if you need more space to write.
```
<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>1F</td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>2A</td>
<td>2B</td>
<td>2C</td>
<td>2D</td>
<td>2E</td>
<td>2F</td>
</tr>
<tr>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>3A</td>
<td>3B</td>
<td>3C</td>
<td>3D</td>
<td>3E</td>
<td>3F</td>
</tr>
<tr>
<td>40</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>4A</td>
<td>4B</td>
<td>4C</td>
<td>4D</td>
<td>4E</td>
<td>4F</td>
</tr>
<tr>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>5A</td>
<td>5B</td>
<td>5C</td>
<td>5D</td>
<td>5E</td>
<td>5F</td>
</tr>
<tr>
<td>60</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>6A</td>
<td>6B</td>
<td>6C</td>
<td>6D</td>
<td>6E</td>
<td>6F</td>
</tr>
<tr>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
<td>7A</td>
<td>7B</td>
<td>7C</td>
<td>7D</td>
<td>7E</td>
<td>7F</td>
</tr>
<tr>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td>86</td>
<td>87</td>
<td>88</td>
<td>89</td>
<td>8A</td>
<td>8B</td>
<td>8C</td>
<td>8D</td>
<td>8E</td>
<td>8F</td>
</tr>
<tr>
<td>90</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td>95</td>
<td>96</td>
<td>97</td>
<td>98</td>
<td>99</td>
<td>9A</td>
<td>9B</td>
<td>9C</td>
<td>9D</td>
<td>9E</td>
<td>9F</td>
</tr>
<tr>
<td>A0</td>
<td>A1</td>
<td>A2</td>
<td>A3</td>
<td>A4</td>
<td>A5</td>
<td>A6</td>
<td>A7</td>
<td>A8</td>
<td>A9</td>
<td>AA</td>
<td>AB</td>
<td>AC</td>
<td>AD</td>
<td>AE</td>
<td>AF</td>
</tr>
<tr>
<td>B0</td>
<td>B1</td>
<td>B2</td>
<td>B3</td>
<td>B4</td>
<td>B5</td>
<td>B6</td>
<td>B7</td>
<td>B8</td>
<td>B9</td>
<td>BA</td>
<td>BB</td>
<td>BC</td>
<td>BD</td>
<td>BE</td>
<td>BF</td>
</tr>
<tr>
<td>C0</td>
<td>C1</td>
<td>C2</td>
<td>C3</td>
<td>C4</td>
<td>C5</td>
<td>C6</td>
<td>C7</td>
<td>C8</td>
<td>C9</td>
<td>CA</td>
<td>CB</td>
<td>CC</td>
<td>CD</td>
<td>CE</td>
<td>CF</td>
</tr>
<tr>
<td>D0</td>
<td>D1</td>
<td>D2</td>
<td>D3</td>
<td>D4</td>
<td>D5</td>
<td>D6</td>
<td>D7</td>
<td>D8</td>
<td>D9</td>
<td>DA</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>DE</td>
<td>DF</td>
</tr>
<tr>
<td>E0</td>
<td>E1</td>
<td>E2</td>
<td>E3</td>
<td>E4</td>
<td>E5</td>
<td>E6</td>
<td>E7</td>
<td>E8</td>
<td>E9</td>
<td>EA</td>
<td>EB</td>
<td>EC</td>
<td>ED</td>
<td>EE</td>
<td>EF</td>
</tr>
<tr>
<td>F0</td>
<td>F1</td>
<td>F2</td>
<td>F3</td>
<td>F4</td>
<td>F5</td>
<td>F6</td>
<td>F7</td>
<td>F8</td>
<td>F9</td>
<td>FA</td>
<td>FB</td>
<td>FC</td>
<td>FD</td>
<td>FE</td>
<td>FF</td>
</tr>
</tbody>
</table>