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 8 questions on 11 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>8</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>10</td>
<td>6</td>
<td>6</td>
<td>32</td>
<td>16</td>
<td>9</td>
<td>12</td>
<td>12</td>
<td>103</td>
</tr>
<tr>
<td>Score:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. (10 points) Define the terms below. You do not need to give a formal definition just a good description. You may include an example if it is helpful to your answer.

(a) compiler

**Solution:** Software which translates a high level language to machine language.

(b) variable

**Solution:** An identifiable memory cell stores information that is necessary to solve a problem.

(c) safe conversion

**Solution:** A conversion from one representation (encoding) to another representation (encoding) where there is no (or very little) loss in accuracy.

(d) infinite loop

**Solution:** An infinite loop is a sequence of instructions in a computer program which loops endlessly, either due to the loop having no terminating condition, having one that can never be met, or one that causes the loop to start over.

(e) escape character

**Solution:** A special character that allow Java to change the meaning of the next character. e.g. \ (backslash) is the escape character for strings.
2. (6 points) Convert the following binary numbers to decimal equivalents
   (a) $10101_2$
   (a) $21$

   (b) $1011_2$
   (b) $11$

3. (6 points) Convert the following decimal number to their binary representation.
   (a) 17
   (a) $10001$

   (b) 45
   (b) $101101$
4. (32 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 write “error” followed by the reason. The first row has been done for you.

```java
String s1 = "Java", s2 = "Computer", s3 = "-42";
char c1 = '5', c2 = '*';
int i1 = -5, i2 = 2, i3 = 6;
double d1 = 4.5, d2 = 9.0;
```

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value (Type)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4+1</td>
<td>5 (int)</td>
</tr>
<tr>
<td>i1 + i2 / i2</td>
<td>-4 (int)</td>
</tr>
<tr>
<td>s1 + i1 * i3</td>
<td>Java-30 (String)</td>
</tr>
<tr>
<td>i1 * d2</td>
<td>-45.0 (double)</td>
</tr>
<tr>
<td>(int)d2 + d1 + i3</td>
<td>19.5 (double)</td>
</tr>
<tr>
<td>i2 != d2 / d1</td>
<td>false (boolean)</td>
</tr>
<tr>
<td>(int)s3 + i2</td>
<td>error (can’t cast from string to int)</td>
</tr>
<tr>
<td>i2 % i3</td>
<td>2 (int)</td>
</tr>
<tr>
<td>d1 / i1 + i2</td>
<td>1.1 (double)</td>
</tr>
<tr>
<td>&quot;&quot; + i3 + c2 + i1</td>
<td>6*-5 (String)</td>
</tr>
<tr>
<td>i1 + s3 + d2</td>
<td>-5-429.0 (String)</td>
</tr>
<tr>
<td>c1 + 2</td>
<td>55 (int)</td>
</tr>
<tr>
<td>s1.charAt(5)</td>
<td>error (index out of range)</td>
</tr>
<tr>
<td>s2.charAt(5)</td>
<td>t (char)</td>
</tr>
<tr>
<td>!(d1 = d2/i2)</td>
<td>error (assignment vs. equality)</td>
</tr>
<tr>
<td>d1 &lt; i2*i3</td>
<td></td>
</tr>
<tr>
<td>i1++ + ++i1</td>
<td>-8 (int)</td>
</tr>
</tbody>
</table>
5. For each of the code fragments below, give the output. If there is no output generated, you may write “none”. If the code results in an infinite loop, write the first few outputs, and then indicate that it is an infinite loop.

(a) (2 points)
```
int x = 1, y = 2, z;
if (x < y) {
    z = x;
    x = y;
    y = z;
}
System.out.println(x);
System.out.println(y);
```

Solution:
2
1

(b) (3 points)
```
int x = 3, y = -2, z = 1;
z += x++ - --y;
System.out.println(x);
System.out.println(y);
System.out.println(z);
```

Solution:
4
-3
7

(c) (2 points)
```
char a = 'C';
switch(a){
    case 'A': System.out.println(a + 1); break;
    case 'B': System.out.println(a); break;
    default: System.out.println(a - 1); break;
}
```

Solution:
66
(d) (2 points)
    String s1 = "Emory", s2 = "University";
    if (s1.charAt(0) <= s2.charAt(0))
        System.out.println(s1.substring(2, s1.length()));
    else
        System.out.println(s2.substring(2, s2.length()));

    Solution:
    ory

(e) (3 points)
    int x = 5;
    while (x != 0) {
        System.out.println(x);
        x -= 2;
    }

    Solution:
    5
    3
    infinite loop!

(f) (4 points)
    for(int x = 2; x <= 11; x++) {
        if (x % 2 == 1)
            continue;
        if (x >= 9)
            break;
        System.out.println(x);
    }

    Solution:
    2
    4
    6
    8
6. Read the following snippet of code and answer the questions below.

```java
String s = "acbadca";
int i;
for (i = 0; i < s.length() / 2; i++) {
    if (s.charAt(i) != s.charAt(s.length() - 1 - i))
        break;
}
if (i == s.length() / 2)
    System.out.println("Yes");
else
    System.out.println("No");
```

(a) (3 points) What’s the output?

**Solution:**

No

(b) (3 points) Change exactly one character in `s` to make the output different. Write down the changed string `s`.

**Solution:**

`acbabc/acdadca`

(c) (3 points) [bonus] Can you find the pattern of string `s` that makes the output “Yes”?

**Solution:**

Palindrome string (a string that reads the same forward or reversed).
7. (12 points) Complete the program below. It should prompt the user to enter 50 integer numbers, and print the largest number to the screen. Assume the input numbers are non-negative.

```java
import java.util.Scanner;

public class PrintMax{
    public static void main (String[] args){
        Scanner in = new Scanner(System.in);
        /*----------- Your code here -------------*/

        Solution:

        // initialize max
        int max = -1, cur;
        // prompt the user to enter 50 numbers
        System.out.println("Please input 50 numbers");
        // find the largest number
        for (int i = 0; i < 50; i++) {
            cur = in.nextInt();
            if (cur > max)
                max = cur;
        }
        System.out.println("The largest number is: " + max);
    }
}
```
8. (12 points) Complete the program below. It should prompt the user to enter three edges for a triangle and determine 1) whether the input is valid (the input is valid when it can form a triangle); 2) whether the triangle is a right triangle when the input is valid. Then it prints the conclusion to the screen. Note that the sum of any two edges must be greater than the third edge in a triangle and that the square of the longest edge equals to the square sum of other two edges in a right triangle.

Sample output:

```java
>>> java Triangle
Please input the edges: 1 2 3
Invalid input.

>> java Triangle
Please input the edges: 1 1 1
Triangle.

>> java Triangle
Please input the edges: 3 4 5
Right triangle.
```

```java
import java.util.Scanner;
public class Triangle {
    public static void main(String[] args) {
        int a, b, c; // variables to store the three edges.
        Scanner in = new Scanner(System.in);
        System.out.print("Please input the edges: ");
        /* ---------------- Your code here ----------------*/
        Solution:
        // read the input edges
        a = in.nextInt();
        b = in.nextInt();
        c = in.nextInt();

        // check input validity
        if (a > 0 && b > 0 && c > 0 && a + b > c && a + c > b && b + c > a) {
            // check right triangle
            if (a * a + b * b == c * c || a * a + c * c == b * b || b * b + c * c == a * a) {
                System.out.println("Right triangle.");
            } else {
                System.out.println("Triangle.");
            }
        } else {
            System.out.println("Invalid input.");
        }
    }
}
```
Solution: Alternative solution.

// read the input edges
a = in.nextInt();
b = in.nextInt();
c = in.nextInt();

// let c store the longest edge.
int t;
if (a > b) {
    if (a > c) {
        t = a;
        a = c;
        c = t;
    }
} else {
    if (b > c) {
        t = b;
        b = c;
        c = t;
    }
}

// check input validity
if (a > 0 && b > 0 && a + b > c) {
    // check right triangle
    if (a * a + b * b == c * c) {
        System.out.println("Right triangle.");
    } else {
        System.out.println("Triangle.");
    }
} else {
    System.out.println("Invalid input.");
}
### ASCII TABLE

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>(NUL)</td>
<td>32</td>
<td>20</td>
<td>(SPACE)</td>
<td>64</td>
<td>40</td>
<td>@</td>
<td>96</td>
<td>60</td>
<td>`</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>(SO)</td>
<td>33</td>
<td>21</td>
<td>!</td>
<td>65</td>
<td>41</td>
<td>A</td>
<td>97</td>
<td>61</td>
<td>a</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>(STX)</td>
<td>34</td>
<td>22</td>
<td>&quot;</td>
<td>66</td>
<td>42</td>
<td>B</td>
<td>98</td>
<td>62</td>
<td>b</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>(ETX)</td>
<td>35</td>
<td>23</td>
<td>#</td>
<td>67</td>
<td>43</td>
<td>C</td>
<td>99</td>
<td>63</td>
<td>c</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>(EOT)</td>
<td>36</td>
<td>24</td>
<td>$</td>
<td>68</td>
<td>44</td>
<td>D</td>
<td>100</td>
<td>64</td>
<td>d</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>(ENQ)</td>
<td>37</td>
<td>25</td>
<td>%</td>
<td>69</td>
<td>45</td>
<td>E</td>
<td>101</td>
<td>65</td>
<td>e</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>(ACK)</td>
<td>38</td>
<td>26</td>
<td>&amp;</td>
<td>70</td>
<td>46</td>
<td>F</td>
<td>102</td>
<td>66</td>
<td>f</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>(BEL)</td>
<td>39</td>
<td>27</td>
<td>'</td>
<td>71</td>
<td>47</td>
<td>G</td>
<td>103</td>
<td>67</td>
<td>g</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>(BS)</td>
<td>40</td>
<td>28</td>
<td>(</td>
<td>72</td>
<td>48</td>
<td>H</td>
<td>104</td>
<td>68</td>
<td>h</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>(HT)</td>
<td>41</td>
<td>29</td>
<td>)</td>
<td>73</td>
<td>49</td>
<td>I</td>
<td>105</td>
<td>69</td>
<td>i</td>
</tr>
<tr>
<td>10</td>
<td>A</td>
<td>(LF)</td>
<td>42</td>
<td>2A</td>
<td>*</td>
<td>74</td>
<td>4A</td>
<td>J</td>
<td>106</td>
<td>6A</td>
<td>j</td>
</tr>
<tr>
<td>11</td>
<td>B</td>
<td>(VT)</td>
<td>43</td>
<td>2B</td>
<td>+</td>
<td>75</td>
<td>4B</td>
<td>K</td>
<td>107</td>
<td>6B</td>
<td>k</td>
</tr>
<tr>
<td>12</td>
<td>C</td>
<td>(FF)</td>
<td>44</td>
<td>2C</td>
<td>,</td>
<td>76</td>
<td>4C</td>
<td>L</td>
<td>108</td>
<td>6C</td>
<td>l</td>
</tr>
<tr>
<td>13</td>
<td>D</td>
<td>(CR)</td>
<td>45</td>
<td>2D</td>
<td>-</td>
<td>77</td>
<td>4D</td>
<td>M</td>
<td>109</td>
<td>6D</td>
<td>m</td>
</tr>
<tr>
<td>14</td>
<td>E</td>
<td>(SOH)</td>
<td>46</td>
<td>2E</td>
<td>.</td>
<td>78</td>
<td>4F</td>
<td>N</td>
<td>110</td>
<td>6E</td>
<td>n</td>
</tr>
<tr>
<td>15</td>
<td>F</td>
<td>(STX)</td>
<td>47</td>
<td>2F</td>
<td>/</td>
<td>79</td>
<td>4G</td>
<td>O</td>
<td>111</td>
<td>6F</td>
<td>o</td>
</tr>
<tr>
<td>16</td>
<td>10</td>
<td>(ESC)</td>
<td>48</td>
<td>30</td>
<td>0</td>
<td>80</td>
<td>50</td>
<td>P</td>
<td>112</td>
<td>70</td>
<td>p</td>
</tr>
<tr>
<td>17</td>
<td>11</td>
<td>(BS)</td>
<td>49</td>
<td>31</td>
<td>1</td>
<td>81</td>
<td>51</td>
<td>Q</td>
<td>113</td>
<td>71</td>
<td>q</td>
</tr>
<tr>
<td>18</td>
<td>12</td>
<td>(HT)</td>
<td>50</td>
<td>32</td>
<td>2</td>
<td>82</td>
<td>52</td>
<td>R</td>
<td>114</td>
<td>72</td>
<td>r</td>
</tr>
<tr>
<td>19</td>
<td>13</td>
<td>(LF)</td>
<td>51</td>
<td>33</td>
<td>3</td>
<td>83</td>
<td>53</td>
<td>S</td>
<td>115</td>
<td>73</td>
<td>s</td>
</tr>
<tr>
<td>20</td>
<td>14</td>
<td>(VT)</td>
<td>52</td>
<td>34</td>
<td>4</td>
<td>84</td>
<td>54</td>
<td>T</td>
<td>116</td>
<td>74</td>
<td>t</td>
</tr>
<tr>
<td>21</td>
<td>15</td>
<td>(FF)</td>
<td>53</td>
<td>35</td>
<td>5</td>
<td>85</td>
<td>55</td>
<td>U</td>
<td>117</td>
<td>75</td>
<td>u</td>
</tr>
<tr>
<td>22</td>
<td>16</td>
<td>(CR)</td>
<td>54</td>
<td>36</td>
<td>6</td>
<td>86</td>
<td>56</td>
<td>V</td>
<td>118</td>
<td>76</td>
<td>v</td>
</tr>
<tr>
<td>23</td>
<td>17</td>
<td>(SU)</td>
<td>55</td>
<td>37</td>
<td>7</td>
<td>87</td>
<td>57</td>
<td>W</td>
<td>119</td>
<td>77</td>
<td>w</td>
</tr>
<tr>
<td>24</td>
<td>18</td>
<td>(SI)</td>
<td>56</td>
<td>38</td>
<td>8</td>
<td>88</td>
<td>58</td>
<td>X</td>
<td>120</td>
<td>78</td>
<td>x</td>
</tr>
<tr>
<td>25</td>
<td>19</td>
<td>(DLE)</td>
<td>57</td>
<td>39</td>
<td>9</td>
<td>89</td>
<td>59</td>
<td>Y</td>
<td>121</td>
<td>79</td>
<td>y</td>
</tr>
<tr>
<td>26</td>
<td>20</td>
<td>(DC1)</td>
<td>58</td>
<td>3A</td>
<td>0</td>
<td>90</td>
<td>5A</td>
<td>Z</td>
<td>122</td>
<td>80</td>
<td>z</td>
</tr>
<tr>
<td>27</td>
<td>21</td>
<td>(DC2)</td>
<td>59</td>
<td>3B</td>
<td>&lt;</td>
<td>91</td>
<td>5B</td>
<td>[</td>
<td>123</td>
<td>81</td>
<td>[</td>
</tr>
<tr>
<td>28</td>
<td>22</td>
<td>(DC3)</td>
<td>60</td>
<td>3C</td>
<td>&gt;</td>
<td>92</td>
<td>5C</td>
<td>\</td>
<td>124</td>
<td>82</td>
<td>\</td>
</tr>
<tr>
<td>29</td>
<td>23</td>
<td>(DC4)</td>
<td>61</td>
<td>3D</td>
<td>?</td>
<td>93</td>
<td>5D</td>
<td>]</td>
<td>125</td>
<td>83</td>
<td>]</td>
</tr>
<tr>
<td>30</td>
<td>24</td>
<td>(NUL)</td>
<td>62</td>
<td>3E</td>
<td>_</td>
<td>94</td>
<td>5E</td>
<td>^</td>
<td>126</td>
<td>84</td>
<td>^</td>
</tr>
<tr>
<td>31</td>
<td>25</td>
<td>(DEL)</td>
<td>63</td>
<td>3F</td>
<td>`</td>
<td>95</td>
<td>5F</td>
<td>~</td>
<td>127</td>
<td>85</td>
<td>~</td>
</tr>
</tbody>
</table>