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 6 questions on 8 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>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>10</td>
<td>20</td>
<td>20</td>
<td>15</td>
<td>20</td>
<td>15</td>
<td>100</td>
</tr>
<tr>
<td>Score:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

1
1. (10 points) For each entry below, state whether the code is correct as written or has an error. Show where the error is (you can circle the incorrect code) and explain why it is an error.

<table>
<thead>
<tr>
<th>Code</th>
<th>Error?</th>
<th>Why?</th>
</tr>
</thead>
<tbody>
<tr>
<td>int[] A={1,2}; int k = A[3];</td>
<td>Yes</td>
<td>Length of array A is 2. Index 3 is out of bounds</td>
</tr>
<tr>
<td>String str = &quot;CD170&quot;; str.charAt(1)=’S’;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>public static boolean L(int k){</td>
<td></td>
<td></td>
</tr>
<tr>
<td>if (k&gt;0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>return 1;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>else</td>
<td></td>
<td></td>
</tr>
<tr>
<td>return -1;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>int[][] A = new int[2][2]; A[2][2]=1;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>String str1 = &quot;Wahr&quot;; char c0 = str1.charAt(0); char c3 = str1.charAt(3); String str3 = c3+&quot; +c0;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>String str = &quot;Rana&quot;; int k=str.length; System.out.print(&quot;Length is &quot;+k);</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. (20 points) Consider the program Parameters which is given below. Assume that the program compiles and runs.

```java
public class Parameters{
    public static int var = 10;
    public static double x = 5.0;
    public static boolean Method1(int k, int[] a){
        int j=k++;
        int[] b = a;
        b[0]=j;
        a[1]=k;
        k+=2;
        Parameters.var--;
        return (a[1]==b[0]);
    }
    public static void main(String[] args){
        int[] array = {1,3,5,7,9};
        int k = array[3];
        Parameters.var+=2;
        if(Method1(k, array))
            System.out.println("Method is true");
        else
            System.out.println("Method is not true");
    }
}
```

1. What type of variable(s) does `Method1` return?

2. What type of variable(s) does `Method1` take as an input?

3. What is (are) the class variable(s)?
4. What would the program print?

5. What is the value of k at the end of the main method?

6. Draw the final version of array at the end of the main method.

7. Draw the final version of the array b at the end of Method1

8. What is the value of var at the end of the main method?
3. (20 points) Consider the following methods.

1. What is each one of these algorithms accomplishing?

   /*---------------Code M1---------------------------*/
   public static int M1(int[] A){
     for(int i=0; i<A.length-1; i++)
       if(A[i]>A[i+1])
         return (i+1);

     return -1;
   }

   /*--------------Code M2----------------------------*/
   public static boolean M2(int[] A){
     boolean flag = true;
     for(int i=0; i<A.length-1; i++)
       if(A[i]>A[i+1]){
         flag = false;
         break;
       }

     return flag;
   }

   /*---------------Code M3---------------------------*/
   public static boolean M3(int[] A){
     for(int i=A.length-1; i>0; i--)
       if(A[i]<A[i-1])
         return false;

     return true;
   }

   /*-----------------Code M4-------------------------*/
   public static boolean M4(int[] A){
     int i=0;
     while(i<A.length-1){
       if(A[i]>A[i+1])
         return false;
       i++;
     }

     return true;
   }

2. Which methods are doing the same thing?
4. (15 points) Draw the array that would result after the following code is executed:

1. int[] A={1,3,4,8,1};
   for (int i=A.length-1; i>=0; i--)
       A[i]=(int)((A[i]-A[0])/2);

2. char[] C = {'A', 'E', 'I', 'O', 'U'};
   char aux = C[0];
   for(int i=0; i<C.length-1; i++)
       C[i]=C[i+1];
   C[C.length-1]=aux;

3. int[] A={1,3,4,8,1,6};
   for (int i=0; i<A.length; i++)
       if(A[i]%2==0)
           A[i]*=2;
       else
           A[i]++;
5. (20 points) Write the method post7. Given a non-empty array of ints, return a new array containing the elements from the original array that come after the last integer '7' in the original array. The original array will contain at least one '7'. Note that it is valid in java to create an array of length 0.

    post7({2, 7, 1, 2}) => {1, 2}
    post7({7, 1, 7, 2}) => {2}
    post7({7, 7, 1, 2, 3}) => {1, 2, 3}
6. (15 points) You are given a 2D Java array with 100 rows and 100 columns

```java
double[][] M = new double[100][100];
```

Suppose that M is already full of double values.

1. Write a method called `getCol` that, given M of any size and an integer k as input, returns a 1D array with the elements of its kth column. You can assume that the number k is always less or equal to the total number of columns.