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 9 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>12</td>
<td>10</td>
<td>20</td>
<td>16</td>
<td>20</td>
<td>22</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. (12 points) For each entry below, state whether the code is correct as written or has an error. If there is no error, write the output. If there is, show where the error is (you can circle the incorrect code) and explain why it is an error.

- Snippet 1

```java
int[] a = {2, 4, 6, 7};
for (int i = 0; i < a.length - 1; i++) {
    System.out.println("Element + 2 = " + (a[i] + 2));
}
```

Output or error:

- Snippet 2

```java
String str = "CS170";
char[] c = str.toCharArray();
for (int i = 0; i < c.length; i++) {
    System.out.println(c[i]);
}
```

Output or error:

- Snippet 3

```java
int[] array = new int[10];
array[9] = 20;
System.out.println((array[0] + array[5] + array[9]));
```

Output or error:

- Snippet 4

```java
public static void main(String[] args){
    String x = "World!";
    {
        int x = 20;
    }
    System.out.println(x);
}
```

Output or error:
1. Snippet 5

```java
int[] a = new int[];
a[0] = 10;
System.out.println((a[0] + 20));
```

Output or error:

2. Snippet 6

```java
public static void cmp(int[] arr1, int[] arr2){
    if (arr1.length == arr2.length) return 1;
    else return -1;
}
public static void main(String[] args){
    int[] ar = {1, 2, 3, 4};
    int[] ar2 = {1, 2, 3};
    int res = cmp(ar, ar2);
    System.out.println(res);
}
```

Output or error:

2. (10 points) Consider the array:

{2, 4, 6, 10, 20, 25, 30, 40, 45, 4, 50, 70}

- (2 points) Is it possible to perform a **linear search** on this array? If not, explain why. If it is possible, search for the value 20 using a linear search and list the elements in order that we will inspect

- (2 points) Is it possible to perform a **binary search** on this array? If not, explain why. If it is possible, search for the value 10 using a binary search and list the elements in order that we will inspect
• (6 points) Consider the array:

\{5, 9, 1, 2, 20, 10, 3\}

Apply the selection sort on this array and show all sorting steps. First step has been done for you

**Step 1:** \{5, 9, 1, 2, 20, 10, 3\}
**Step 2:** \{1, 9, 5, 2, 20, 10, 3\}

3. (20 points) Draw the array \( arr \) after the code below executes. Assume that each code snippet is executed separately.

• Snippet 1

```java
int [] arr = {1, 4, 7, 10, 13};
for (int i = 0; i < arr.length - 1; i++) {
    arr[i] = arr[i+1] * 2;
}
```

• Snippet 2

```java
int [] a = {4, 0, 20, 100, -40};
int [] b = {-20, 0, 10, 2, 1};
int [] arr = new int[a.length];
for (int i = 0; i < arr.length; i++){
    if (a[i] == b[i])
        arr[i] = 0;
    else if (a[i] > b[i])
        arr[i] = 1;
    else if (a[i] < b[i])
        arr[i] = -1;
}
```
• Snippet 3

```java
int [][] A = {{1, 2, 3, 4},
              {5, 6, 7, 8},
              {9, 0, 1, 2}};

int [] arr = new int [A.length];
for (int i = 0; i < A.length; i++) {
    int sum = 0;
    for (int j = 0; j < A[i].length; j++) {
        sum += A[i][j];
    }
    arr[i] = sum;
}
```

• Snippet 4

```java
int [] A = {1, 3, 5, 10, 20};
int [] B = {0, 20, 5, 7, 30};
int [] arr = new int [A.length + B.length];

int j = 0;
for (int i = 0; i < A.length; i++) {
    if (A[i] > B[i]) {
        arr[j++] = B[i];
        arr[j++] = A[i];
    }
    else {
        arr[j++] = A[i];
        arr[j++] = B[i];
    }
}
```
4. (16 points) Consider the following program:

```java
public class Exam1 {
    public static int x = 10;

    public static void resInt(int x, int val) {
        System.out.println(x); // Position 3
        x = val;
    }

    public static void setInt(int y) {
        System.out.println(x); // Position 6
        System.out.println(Exam1.x); // Position 7
        System.out.println(num); // Position 8
        y = 0;
    }

    public static void main(String[] args) {
        int num = 100;
        String str = "Happy world!";
        System.out.println(x); // Position 1
        System.out.println(str); // Position 2
        resInt(num, 20);
        System.out.println(num); // Position 4
        {
            String CS = "Computer Science";
            System.out.println(str); // Position 5;
        }
        setInt(num);
        System.out.println(CS); // Position 9
        System.out.println(Exam1.x); // Position 10
    }
}
```

1. (10 points) Determine what would be printed in the following positions. If a variable cannot be printed, write SCOPING ERROR.

- Position 1:
- Position 2:
- Position 3:
- Position 4:
- Position 5:
- Position 6:
- Position 7:
- Position 8:
- Position 9:
- Position 10:
2. (2 points) What line number is the class variable defined on? What is the scope of the class variables?

3. (2 points) What type of variable(s) does the method `restInt` take as an input? (parameters)

4. (2 points) What type of variable does `setInt` return?

5. (20 points) Write the following methods:
   - Write a method called `oddsAverage` that, given an array of integers `arr` will return an average (double) of all odd numbers in this array. For example:
     - `oddsAverage({1, 7, 10, 20, 5})` returns `4.33 ((1 + 7 + 5) / 3.0 = 4.33),`
     - `oddsAverage({1, 3, 7, 9, 5})` returns `5.0 ((1 + 3 + 7 + 9 + 5) / 5.0 = 5.0)"
• Write a method called `betweenRange` that, given an array of integers `arr` and two integers: `start` and `end` will return an integer that is the number of elements in the array that are in the range `<start; end>` (greater than or equal to `start` and less than or equal to `end`).
For example:
`betweenRange({1, 4, 10, 20, 5}, 2, 10)` returns 3 (elements 4, 10 and 5 in `<2, 10>`),
`betweenRange({1, 4, 7, 8}, 10, 12)` returns 0 (no elements in `<10; 12>`).
6. (22 points) Write a method called `largestAverage` that, given a two dimensional array `arr` will return a row index with the largest average of elements. For example, if `arr` is:

\[
\begin{bmatrix}
2 & 5 & 1 & 7 \\
10 & 0 & 1 & 2 \\
5 & 6 & 7 & 1 \\
10 & 4 & 2 & 1 \\
\end{bmatrix}
\]

the method should return 2 (row 2: \(\frac{5 + 6 + 7 + 1}{4} = 4.75\)), which has the largest average of all rows.