This exam is closed-books/notes/computers, closed-internet, closed-calculators, closed-devices, and closed-neighbors. Show your work in each problem for full credit.

Name:

____________________________________________________________________

Honor Code:
Q1. Short Answers (6 points)

(a) Given an array, myArray, give the first and last valid indices (indexes).

first valid index: _________________________
last valid index: _________________________

(b) What are the values of t and k right after t is updated in the for loop? How many times does the body of the loop run?

```java
int[] array = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19};
int t = 100;
int k;
for (int j = array.length-1; j >=0; j = j - 3) {
    k = array[j];
    t = t - k;
    // values of t and k here
}
```

# of times the for loop body is executed: _______________

<table>
<thead>
<tr>
<th>Iteration 1</th>
<th>Iteration 2</th>
<th>Iteration 3</th>
<th>Iteration 4</th>
<th>Iteration 5</th>
<th>Iteration 6</th>
<th>Iteration 7</th>
<th>Iteration 8</th>
<th>Iteration 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>t</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question</th>
<th>Max</th>
<th>Your score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>58</td>
<td></td>
</tr>
</tbody>
</table>
Q2. Code Trace (10 points)

(a) Fill as many as rows as necessary in the table below to trace the loop and convert the for loop to a while loop. Write only the necessary new code and strike out the code to be removed. Clearly indicate where the new code should be placed.

String[] words = { “excellence”, “is”, “not”, “a”, “skill”, “it”, “is”, “an”, “attitude”};

String verdict = “”;
for (int i = 0; i < 9; i = i + 1) {
    String w = words[i];
    int wn = w.length();
    char ch = w.charAt(wn - 1);

    if (strContains(“facetious”, ch) == false) {
        verdict = verdict + “0”;
    } else {
        verdict = verdict + “1”;
    }

    // fill one row in the table at this point
}

<table>
<thead>
<tr>
<th>i</th>
<th>w</th>
<th>wn</th>
<th>ch</th>
<th>verdict</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
(b) Fill as many as rows necessary in the table right after the String s is assigned \textbf{and convert} the \textit{while loops} to \textit{for loops}. You may write only necessary parts next to the presented code but make clear \textit{what} and \textit{where} the changes are.

\begin{verbatim}
int k = 0;
int i = 15;
while (i > 8) {
    int j = 1;
    while (j < i/2) {
        if (i % j != 0) {
            j = j + 2;
        }
        k = i + j;
        j = j + 1;
    // fill one row at this point
    }
    i = i - 3;
}
\end{verbatim}

<table>
<thead>
<tr>
<th>i</th>
<th>j</th>
<th>k</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Q3. Arrays and Loops (10 points)

(a) [while loop] Write a method, `pairAvg`, which takes a non-empty `double` array and returns a new `double` array containing pairwise averages. Assume that the input array has even number of elements.

E.g.,
```java
double[] a2 = { 10, -30, 20, 0, 40, 50 };
double[] a3 = { 60, 80 };
pairAvg(a2) returns {-10, 10, 45}.  // -10 = average of 10 and 30, 10 = average of 20 and 0, ...
pairAvg(a3) returns {70}.  // 70 = average of 60 and 80
```

(b) [for loop] Write a method, `allMultipleBase`, which takes an integer array `array` and a positive integer `base` and returns true if all elements in the array are multiples of `base`. No nested loop is allowed.

E.g.,
```java
int[] a4 = { 6, 9, 256, 273 };
allMultipleBase(a4, 3) returns true.
allMultipleBase(a4, 6) returns false since 9 and 273 are not multiples of 6.
```
Q4. Strings and Loops (11 points)

(a) [while loop] Write a method, `extractConsonants`, which takes a String returns a new String with only the consonants from the input String. You may use `strContains` method from class.

E.g., `extractConsonants("FACETIOUS")` returns “FCTS”
`extractConsonants("MISSISSIPI")` returns “MSSSSP”

(b) [for loop] Write a method, `bin2dec`, which takes a String of 1’s and 0’s and returns an equivalent decimal number as an integer. Examine each character of the String and if it is ‘1’, add an assigned 2’s power. From the right-most digit, the assigned 2’s powers are $2^0$, $2^1$, $2^2$, $2^3$, etc. You may NOT use Math.pow(..) method or write another loop inside or outside of the main loop to compute 2’s powers.

E.g., `bin2dec("10100")` returns 20 ($2^4 + 2^2 = 16 + 4$)
`bin2dec("1001")` returns 9 ($2^3 + 2^0 = 8 + 1$)
Q5. Nested Loops (21 points)

(a) [for loops] Write a method, `drawLRT(String s1, String s2, int n, int x0, int y0)`, which draws a lower-right triangular shape with alternating rows of s1’s and s2’s up to n rows. The rows and columns are separated by 25 pixels and the first String is drawn at (x0, y0). You must use nested for loops for full credit. Use `canvas.drawText(x, y, text);` to draw text at (x, y).

E.g., `drawLRT("@", "?", 6, 200, 100);`

```
75 100 125 150 175 200
@  100
? ? 125
@  @  @  150
@  @  @  @  200
```
(b) Write a method, \texttt{incString(String s)}, which returns an array of String with each element of the array containing one more letter of \texttt{s} than the previous element. You may only use \texttt{charAt(.)} and \texttt{length()} methods from \texttt{String} and must use \textit{nested loops} for full credit.

E.g., \texttt{incString(“HELLO”)} returns \{ “H”, “HE”, “HEL”, “HELL”, “HELLO” \}. 
Write a method, `drawHourGlass(int x0, int y0, int n)`, which draws an hour-glass shape using asterisks. (x0, y0) is the location for the *top-left-most* asterisk and n is the total number of rows. Assume n is odd.

```
drawHourGlass(200, 100, 3);

* * *
*  *
* * *

drawHourGlass(200, 100, 5);

* * * *
*  *  *
*  *
* * * *
* * * * *
```