1. What gets printed? Please show output as it will appear or indicate “NO OUTPUT” or show some of the output followed by “INFINITE LOOP.”

```java
int a = 4;
while (a < 8) {
    a++;
    System.out.println(a);
}
```

Output: 5 6 7 8

```java
int a = 4;
while (a < 8) {
    System.out.println(a);
    a--;
}
```

Output: 4 3 2 ...

```java
int a = 4;
while (a <= 5) {
    System.out.println(2*a);
    a++;
}
```

Output: 8 10

```java
int a = 4;
while (a < 14) {
    System.out.println(a);
    a = a + 3;
}
```

Output: 4 7 10 13

CSC1051 Data Structures and Algorithms I

Dr. Papalaskari

Spring 2014
1. What gets printed? Please show output as it will appear or indicate “NO OUTPUT” or show some of the output followed by “INFINITE LOOP.”

```java
int a = 4;
while (a > 0) {
    System.out.println(a);
    a++;
}
```

Output: 4 5 6 ... INFINITE LOOP

```java
int a = 4;
while (a > 0) {
    System.out.println(a);
    a = a - 2;
}
```

Output: 4 2

```java
int a = 1;
while (a < 4) {
    a++;
    System.out.println(a);
}
```

Output: 2 3 4

```java
int a = 4;
while (a < 4) {
    System.out.println (a);
    a++;
}
```

Output: NO OUTPUT
2. Let’s look at the problem of repeatedly obtaining input and performing a calculation, for example, computing the circumference of a circle given its radius, using the following algorithm:

<table>
<thead>
<tr>
<th>Variables:</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius, circ</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm:</th>
</tr>
</thead>
<tbody>
<tr>
<td>input radius</td>
</tr>
<tr>
<td>circ = 2 * radius* PI</td>
</tr>
<tr>
<td>print circ</td>
</tr>
</tbody>
</table>

*Rewrite this algorithm, modifying it so that it uses a while structure to repeat the processing of each input in two different ways.*

a) Keep computing circumferences and **ask each time** whether to keep going.

<table>
<thead>
<tr>
<th>Variables:</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius, circ, ans</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ans = 1</td>
</tr>
<tr>
<td>while (ans equals 1)</td>
</tr>
<tr>
<td>input radius</td>
</tr>
<tr>
<td>circ = 2 * radius* PI</td>
</tr>
<tr>
<td>print circ</td>
</tr>
<tr>
<td>print “do another?”</td>
</tr>
<tr>
<td>input ans</td>
</tr>
</tbody>
</table>

b) Compute the circumferences of 10 circles (**exact count**).

<table>
<thead>
<tr>
<th>Variables:</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius, circ, count</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm:</th>
</tr>
</thead>
<tbody>
<tr>
<td>count= 1</td>
</tr>
<tr>
<td>while (count &lt;= 10)</td>
</tr>
<tr>
<td>input radius</td>
</tr>
<tr>
<td>circ = 2 * radius* PI</td>
</tr>
<tr>
<td>print circ</td>
</tr>
<tr>
<td>count = count + 1</td>
</tr>
</tbody>
</table>
2. Let's look at the problem of repeatedly obtaining input and performing a calculation, for example, computing the area of a circle given its radius, using the following algorithm:

```
Variables:
  radius, area

Algorithm:
  input radius
  area = radius * radius * PI
  print area
```

Rewrite this algorithm, modifying it so that it uses a while structure to repeat the processing of each input in two different ways.

a) Compute the areas of 5 circles (exact count).

```
Variables:
  radius, area, count

Algorithm:
  count = 1
  while (count <= 5)
    input radius
    area = radius * radius * PI
    print area
    count = count + 1
```

b) Keep computing circle areas until user inputs -1 for the radius (sentinel value).

```
Variables:
  radius, area

Algorithm:
  input radius
  while (radius != -1)
    area = radius * radius * PI
    print area
    input radius
```