



Topic: For, while and do-while loops

Activity Guidelines

Group Size: 3

Method of Assigning Students: Count the number of students in the class, divide by 3, count off from 1 to the quotient, and group identical numbers.

Materials:

- ✓ Handout (one copy per group) with questions to be answered at the end of the session

Roles:

Coordinator/Leader: Clarifies goals and objectives, allocates roles for each team member and divides the tasks within the group.

Monitor/Evaluator: Person designed to evaluate the different ideas to approach the problem and make an accurate judgment of the most beneficial option.

Implementer: Person in charge to transform discussions and ideas into a technical solution for the given problem.

Individual Accountability: Every student will be working on teams for the exercises. Every team member is assigned a specific role that will allow everyone on the group to get involved and participate in the problem-solving activity. Every team member should participate in solving the exercises according to their determined role in the group.

Activity Summary

Create a program to add up numbers sequentially using loops to avoid making every single operation individually. Trace through 5 different types of code involving loops to solve for the output. Every student should be able to develop concise code by making use of the different types of loops to come out with a solution for problems that tend to behave in a repetitive manner.



INTRODUCTION TO COMPUTER SCIENCE

PEER SESSION

Do, while and do-while loops

1. Write a program called **SumAndAverage** to produce the sum of 1, 2, 3... to an upperbound (e.g., 100) using a for or while loop. The output shall look like:

The sum is 5050

```
public class Sum {
    public static void main (String[] args) {
        int sum = 0;
        int lowerbound = 1;    // the lower bound to sum
        int upperbound = 100; // the upper bound to sum

        for (int number = lowerbound; number <= upperbound; ++number) { // for
loop
            sum += number;    // same as "sum = sum + number"
        }
        System.out.println("The sum between the numbers " + lowerbound + " and "
+ upperbound + " is: " + sum);
    }
}
```

2. For the next exercises, indicate the output that will be produced. Assume the following declarations are made just before each exercise. That is, assume these initializations are in effect at the beginning of each problem:

Final int MAX = 20; int num = 15;

a.

```
while (num < MAX){
    System.out.println (num);
    num = num + 1;
}
```

Output: 15 16 17 18 19

b.

```
do {
    num = num + 1;
    System.out.println (num);
} while (num <= MAX);
```

Output: 16 17 18 19 20 21



```
c. while (num < MAX) {  
    System.out.println (num);  
    num = num - 1;  
}
```

Output: 15 14 13... *infinitely*

```
d. for (int value=0; value >= 7; value++)  
    System.out.println (value);
```

Output: Does not work

```
e. for (int value=1; value <= 20; value+=4)  
    System.out.println (value);
```

Output: 1 5 9 13 17

