

Please complete the textbook sections through 1.9 before beginning this assignment.

CS 278 and CS 465 PA 3 : Quantified Statements

All programs you submit in this class must follow the Documentation and Style Guidelines.
This document can be found in the Canvas Modules.

All programs you submit in this class must compile with the Oracle Java compiler on the Linux machines in SH 118 or SH 118B.

Review of Quantified Statements

Recall that if P is a predicate and the domain of P is a finite set of elements $\{a_1, a_2, \dots, a_k\}$, then:

$$\forall x P(x) \equiv P(a_1) \wedge P(a_2) \wedge \dots \wedge P(a_k)$$

$$\exists x P(x) \equiv P(a_1) \vee P(a_2) \vee \dots \vee P(a_k)$$

Hint: You will find the Identify Laws very helpful.

$$F \vee p \equiv p \qquad T \wedge p \equiv p$$

The Big Picture

For this lab you will write a Java program that inputs ten different integers and stores them in an array of length 10. (Note: There cannot be duplicates in the array. You may assume that the user does not type any duplicates.)

The domain D is the set of elements in this array.

Your program should evaluate the 4 statements shown below, with respect to the given domain, and determine whether they are true or false.

For each statement your program will print the statement label (a, b, c or d) and the truth value (boolean).

(Make sure that the methods in PA2Methods.java work correctly before proceeding with this assignment.)

Statements:

- a) $\forall x (\text{if } x < 0 \text{ then } x \text{ is even})$
- b) $\exists x (\text{if } x < 0 \text{ then } x \text{ is even})$
- c) $\exists x (x < 0 \text{ and } x \text{ is even})$
- d) $\forall x ((x - 1) \% 3 == 0 \text{ or } (x - 1) \% 2 == 0)$

Example Program Run: (user input is shown in green, output is shown in blue)

Please enter 10 different integers: 3 5 -3 6 0 -5 1 7 -1 8

a) is false
b) is true
c) is false
d) is false

Note: There is no typo in the sample output. The answer to b) is true. The first array element is 3. Let's use 3 as x. $3 < 0$ is false. \therefore Statement b "if $3 < 0$ then 3 is even" is true.

- Read the software quality requirements on page 3 before you proceed.
- If you haven't already done so, read the Documentation and Style Requirements. (Canvas --> Modules --> Resources).

Steps

1. Create a new folder for PA 3.
2. ~~Copy the PA2Methods.java file from PA 2 into your folder for PA 3. (Keep the filename PA2Methods.java. Do not change the filename.)~~
Download Dr. Steiner's PA2Methods.java file and store it in your folder for PA 3.
3. Create a new file called PA3.java.
4. Add header comments.
5. Write a public, static method called **statementA** that will accept an integer array as its parameter. Place this method ABOVE the main method.

The method should evaluate Statement a) from page 1 and return its truth value (as a boolean).

- You must use a "for-each" loop that iterates over the integer array.
- You must call the appropriate methods from PA2Methods.java whenever the statement calls for a logical operation (not, and, or, implies).
- You may write a return statement inside the loop.
- You may not use the number 10 in this method.
- You may not write a break statement.

6. In the main method:
 - Declare the Scanner.
 - Declare an integer array and allocate space for 10 elements.
 - Fill the array with input from the user. One prompt (as shown above) is sufficient. You may assume the user always enters 10 integers with no duplicates. You do not have to write code to check for duplicates.
 - Call the statementA method. Print the boolean value returned by the method (see the example output above for the format).
 - Compile. Test. Debug. Test. Repeat.

7. Write a public, static method called **statementB** that will accept an integer array as its parameter. Place this method ABOVE the main method but BELOW the statementA method.

The method should evaluate Statement b) from page 1 and return its truth value (as a boolean).

- You must use a "for-each" loop that iterates over the integer array.
- You must call the appropriate methods from PA2Methods.java whenever the statement calls for a logical operation (not, and, or, implies).
- You may write a return statement inside the loop.
- You may not use the number 10 in this method.
- You may not write a break statement.

8. Add code to the main method to test statementB. Compile. Test. Debug. Test. Repeat.
9. Repeat the process for statementC. Place the statementC method ABOVE the main method but BELOW the statementB method.

Note: The 5 bulleted code requirements apply to statementC, also.

10. Repeat the process for statementD. Place the statementD method ABOVE the main method but BELOW the statementC method.

Note: The 5 bulleted code requirements apply to statementD, also.

11. Once you have verified that your program output is correct for the example input, run the program with a different set of integers.

Software Quality Requirements

Modifiability: The number 10 can only appear one time in the program, at the point where you instantiate the array. (This makes it easy to change the program to work with a different size array.)

Code reuse: You already have the methods from PA 2 (not, or, and, implies). Call the appropriate method(s) from PA2Methods.java when the logical operation appears in a predicate or in the evaluation of a quantified statement. Do not rewrite the logical operator methods.

You already have the methods booleanToChar and charToBoolean methods that are needed to convert from one type to the other. Do not rewrite the conversion methods.

Submit PA3.java on Canvas.