

Lesson 4: Logical Operators

45 minutes

Overview

How can logical operators be used to evaluate multiple conditions?

Students revisit logical operators to explore the use of the `&&` (and) and `||` (or) operators. Students use truth tables to evaluate Boolean expressions and learn about short-circuited evaluation.

Standards

Full Course Alignment

CSA Conceptual Framework

- **CON-1** - The way variables and operators are sequenced and combined in an expression determines the computed result.

Agenda

Warm Up (10 minutes)

CS Bingo

Activity (30 minutes)

Logical Operators

Evaluating Compound Boolean Expressions

Wrap Up (5 minutes)

Software Engineering Skills

Assessment: Check for Understanding

AP Classroom Topic Questions

Objectives

Students will be able to:

- Construct a conditional statement involving logical operators
- Explain the effect of logical operators `!` (not), `&&` (and), and `||` (or)

Preparation

- Print copies of the Truth Tables handout (one for each student)
- Print copies of the CS Bingo Cards (one card for each student)
- Print the CS Bingo Definitions resource and mix them up
- Gather sticky notes or scrap pieces of paper (one for each student)
- Check the **Teacher's Lounge** for verified teachers on the CSA Forum to find additional strategies or resources shared by fellow teachers

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the students

- **CS Bingo Cards** - Handout
- **Truth Tables** - Handout
- **U4L4 Extra Practice** - Handout

Vocabulary

- **compound Boolean expression** - an expression using logical

operators that evaluates to a Boolean value

- **short-circuited evaluation** - a process in which the evaluation of a logical expression exits when the result is clear, even before the complete evaluation of the expression
- **truth table** - a table used to determine the truth values of a Boolean expression

Teaching Guide


Warm Up (10 minutes)


CS Bingo

Remarks

We have learned a lot of new terms so far! Let's review some of these terms through a game of CS Bingo.

 **Distribute:** Give each student a unique CS Bingo Card.

 **Do This:** Review the instructions for playing CS Bingo.

 **Do This:** Play the music clip to cue the CS Bingo activity. Choose a random definition and read it aloud to the class. Give students a moment to check their bingo card for the correct term, then state the correct term. Repeat until a student yells "Java."

Teaching Tip


To allow for multiple students to "win" the game, continue pulling definitions until a couple more students win as well.

Activity (30 minutes)


Logical Operators (15 minutes)

Remarks

We previously learned to use the NOT operator to negate a Boolean expression. We have also learned to use nested `if` statements to check multiple conditions before executing a block of code. There are more logical operators that we can use to check multiple conditions in one line.


 **Do This:** Review the lesson objectives.


Group: Place students in pairs.


 **Do This:** Direct students to Level 1 on Code Studio to investigate the program with a partner. Students make the changes to the program as prompted.

1

Investigate: Logical Operators


 **Do This:** Click through the animated slide to introduce the logical operators.

 **Do This:** Click through the animated slide to define and demonstrate compound Boolean expressions.

 **Discuss:** Click through the animated slide to display the prompts. Use the Retrieve-Pair-Share strategy to discuss the prompts.

- *How might logical operators help us simplify the use of conditional statements?*
- *In the previous lesson, we used nested `if` statements to solve this scenario. How would this solution change using logical operators?*

Discussion Goal: Students suggest using logical operators instead of multiple or nested `if` statements to check whether multiple conditions are `true` or if at least one of two conditions are `true`. Students share examples of how they would rewrite the code they wrote in the previous lesson to use logical operators instead of nested `if` statements.

 **Do This:** Explain the example conditional statement using a logical operator.

Evaluating Compound Boolean Expressions (15 minutes)


Remarks

Compound Boolean expressions can get pretty complex and sometimes are tricky to evaluate. We can use a tool called a truth table to help us evaluate compound Boolean expressions.

 **Do This:** Define *truth table*.

Remarks

Even though compound Boolean expressions can get complex, every compound Boolean expression can be broken down into our three core logical operators: not, and, and or.

 **Do This:** Click through the animated slide to demonstrate the truth table for `A || !B`.

 **Do This:** Define *short-circuited evaluation*.

 **Do This:** Direct students to complete the Truth Tables handout.

Wrap Up (5 minutes)

Software Engineering Skills

Remarks

You have learned a lot about how software engineers use conditionals and logical operators in their programs! Let's take a moment to reflect on your progress in developing your software engineering skills.

 **Do This:** Have students respond to the reflection prompts on the Unit 4 Guide.

 **Do This:** Review the concepts covered in this lesson.

 **Display:** Key Vocabulary

Assessment: Check for Understanding

Check For Understanding Question(s) and solutions can be found in each lesson on Code Studio. These questions can be used for an exit ticket.



Check for Understanding

AP Classroom Topic Questions

To assign questions from the AP Classroom Question Bank that align with this lesson, create a custom quiz in AP Classroom by searching the Question Bank for the Essential Knowledge statements listed at the top of this lesson plan. You can find instructions and video demonstrations to do this on **AP Central**.

The following Topic Questions in AP Classroom can be assigned as a formative assessment for this lesson:

- Topic Questions 3.5

Note: *Some Learning Objectives and Essential Knowledge statements in the suggested Topic Questions are covered in later units.*



This curriculum is available under a
Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes **contact us**.