Make a copy of this Google Doc template and save the copy in your Google Drive.

As you read the following OLI pages and complete the interactive activities, capture the screenshots of the completed activities and replace the respective screenshots in the document.

- Page 21 Choose from two options
- Page 22 Choose from multiple options
- Page 23 Logic/Boolean operators

When you are ready to submit the assignment, download the document in PDF and submit the PDF file on Cougar Course as the proof for your work.

# Page 21 Choose from two options (Part II)

# LBD calling absolute

Mark the function calls that would trigger the execution of the <i>if-block</i> in the absolute
function.
absolute(-28)
☑ absolute(0)
absolute(-4.1)
✓ absolute(2.5)
Check My Answer
Correct! The if-block is executed when the parameter receives a non-negative value as its argument.
Complete the following function call with a value that would trigger the execution of the else-block (line 5) in the absolute function.
absolute( ]-1 );
Correct! The else-block is executed when the parameter receives a negative value as its argument.

### LBD if without else

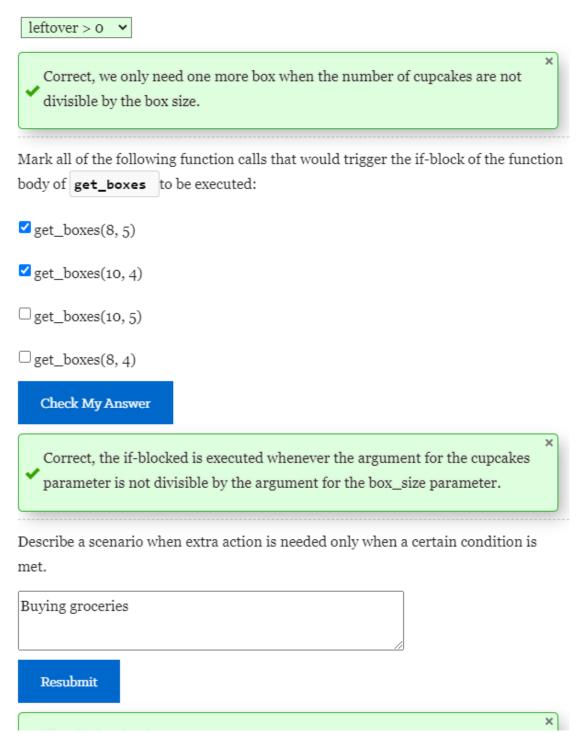
Mark all of the following function calls that would trigger the *if-block* of the function body of <code>final\_cost</code> to be executed:

- ✓ final\_cost(55)
- final\_cost(45)
- ✓ final\_cost(50)

### Check My Answer

Correct, the if-blocked is executed whenever the purchase parameter takes on an argument that is 50 or more.

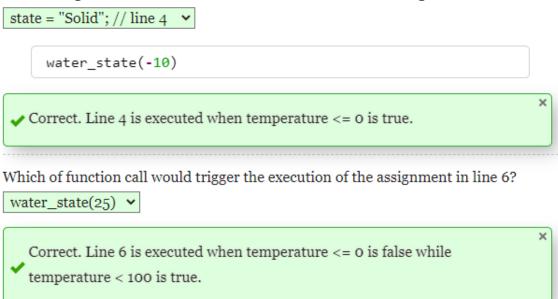
Enter a number in the following function call that would trigger the if-block in the function body to be executed:



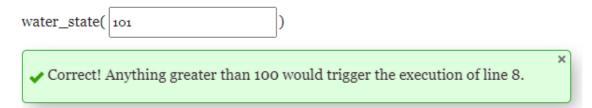
## Page 22 Choose from multiple options

### LBD calling water\_state

Which assignment statement will be executed with the following function call?



Enter an argument for the following function call in order for the return value of the function to be "Gas":



Mark all comparisons the compiler will need to evaluate before returning 'C' for the function call letter\_grade(75).

- ☐ grade >= 60
- ✓ grade >= 90
- ✓ grade >= 80

#### Check My Answer

Correct, the compiler will first evaluate grade >= 90, then grade >= 80, and finally grade >= 70. grade >= 60 will not be evaluated since grade >= 70 is true.

In your own word, explain why the function call letter\_grade(75) would not return 'D' even though grade >= 60 is true.

Because 75 is greater than or equal to 70 which outputs the letter C.

#### Resubmit

→ Thanks for sharing.

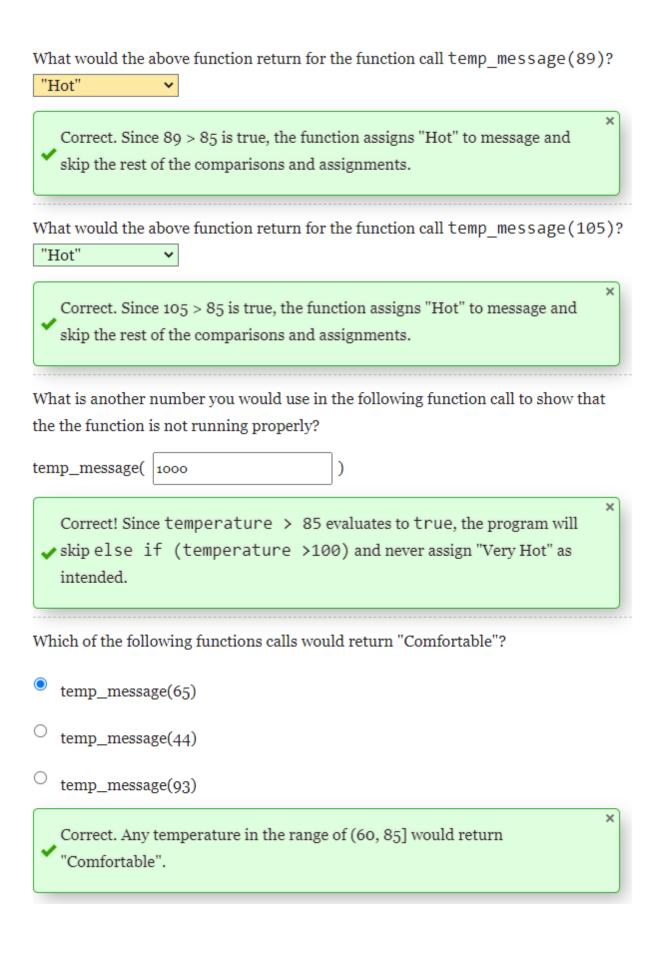
What would be the problem if we change the function, using if instead of else if for the comparisons after grade >= 90? Click on this link to test the function in RepLit.

```
char letter_grade(int grade) {
    char letter;
    if (grade >= 90) {
        letter = 'A';
    }
    if (grade >= 80) {
        letter = 'B';
    }
    if (grade >= 70) {
        letter = 'C';
    }
    if (grade >= 60) {
        letter = 'D';
    }
    else {
        letter = 'F';
    }
    return letter;
}
```

It uses the last if statement as the main so any input greater than or equal to 60 will output D, even if its 70+

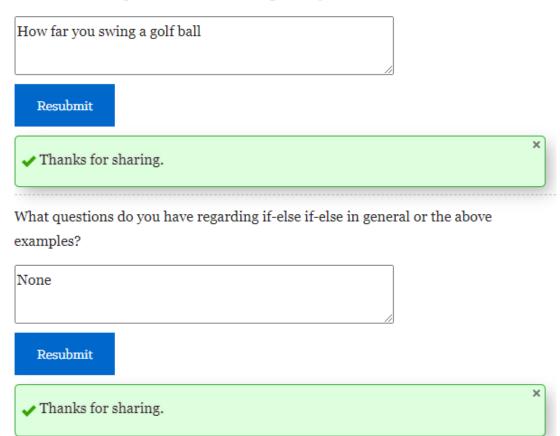
Resubmit

LBD temperature message



# MR consecutive range examples

What other examples of consecutive ranges can you think of?



### Checkpoint switch statement:

Copy and Paste a screenshot of your expanded switch statement to replace the following.

- 1. Make sure you are logged in with your Repl.it account.
- 2. Click this link to access a program that contains the above ice cream flavor survey.
- 3. Add your name and section on the top of the program.
- Expand the switch statement to include additional cases so that each ice cream flavor option will receive its own feedback.

When you are done, copy and paste the switch statement into the following box.

```
switch (response) {
case 'A':
case 'a':
cout << "Great, that's my favorite too!" << endl;
break;
case 'B':
case 'b':
cout << "Good choice!" << endl:
break:
case 'C':
case 'c':
cout << "Yummy!" << endl;\
break;
case 'D':
case 'd':
cout << "Flavorful Fruit!" << endl;
 cout << "Sorry, not a valid option." << endl;
```

Resubmit

✓ Thanks for sharing.

Describe a scenario where different actions would be performed based on specific characters or int values.

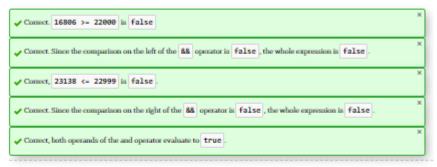


## Page 23 Logic/Boolean operators

### LBD logic operator &&

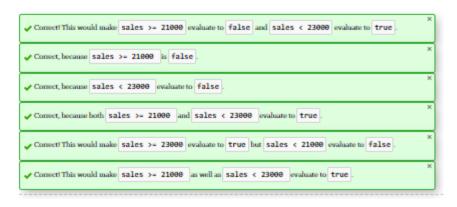
Choose appropriate values for the following conditions using data from the above table.





Enter numbers of your choice to the first column of the following table to generate the results in the second and third columns. Choose the right value for the last column.





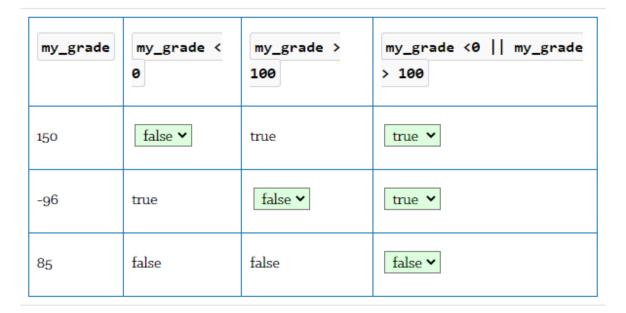
The above function is set up to return true if hour is within the range of 1 and 12. In the following space, write the conditional expression using &&.

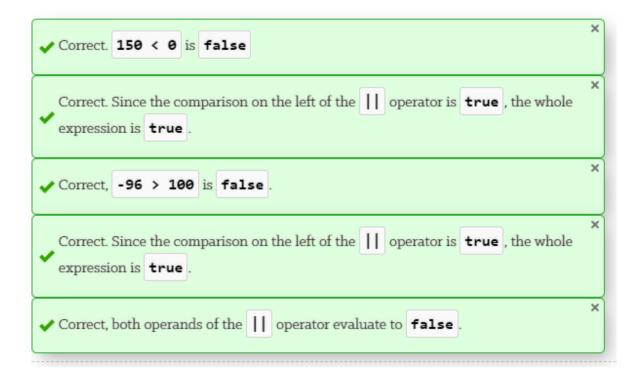
if (valid >= 1 && valid <= 12)

Resubmit

LBD logic operator ||

Choose appropriate values for the following conditions based on the value of my\_grade in the first column.





In your own words, explain why it is not a good idea to use (my\_grade < 0 && my\_grade > 100) as a conditional expression.

Because it is impossible to have a number less than 0 and greater than 100

#### Resubmit

This would require my\_grade to be both negative and larger than 100, which is impossible.

In your own words, explain why it is not a good idea to use (my\_grade >= 0 || my\_grade <= 100) as a conditional expression.

Because the condition will always be true

#### Resubmit

This condition will never become false.

×

The above function is set up to return false if hour is outside the range of 1 and 12.
Note the if-block sets valid to false and the else-block sets valid to true.
In the following space, write the conditional expression using $\hfill \square$ .
if (valid < 1    valid > 12)
Resubmit
Do you have a preference between using && to check for within the range and using
check for outside the range?
Yes, I prefer    to check
Resubmit
✓ Thanks for sharing.

## LBD logic operator!



# LBD Magic 8 ball

The find method returns npos if the given substring is not found in the invoking object. That is, question.find("ball") will return npos if "ball" is not found in question.
<pre>question.find("ball") != npos &amp;&amp; question.find("we") != npos</pre>
Suppose the magic_8 function now uses the above for its condition expression. Mark all function calls that would return "Yes".
☑ magic_8("Can we play a ball game?")
□ magic_8("Can we play the Magic 8 game?")
✓ magic_8("Are we buying a Magic 8 ball app?")
□ magic_8("Do I need a Magic 8 ball app?")
Check My Answer
✓ Correct, both of these questions contain "we" and "ball".
<pre>question.find("ball") != npos    question.find("we") != npos</pre> <pre></pre>
Suppose the magic_8 function now uses the above for its condition expression. Mark all function calls that would return "Yes".
✓ magic_8("Can we play the Magic 8 game?")
☑ magic_8("Can we play a ball game?")
☑ magic_8("Do I need a Magic 8 ball app?")
✓ magic_8("Are we buying a Magic 8 ball app?")
Check My Answer
✓ Correct, all of these questions contain either "we" or "ball".
In the following space, create your own conditional expression to replace the one in magic_8. Be sure to include at least one logic operator.
answer.find("May" != npos    answer.find("please") != npos
Resubmit
▼ Thanks for sharing.