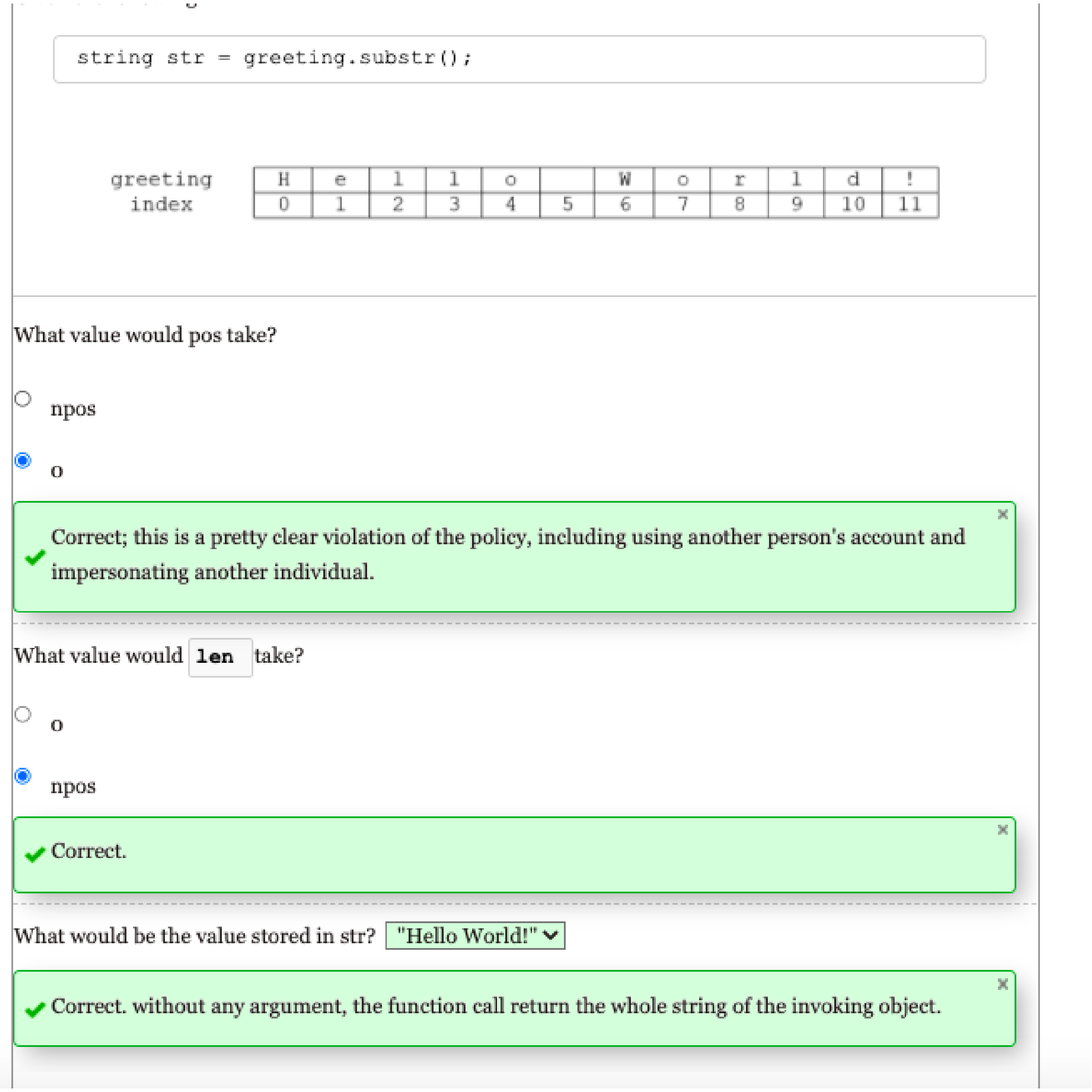


```
string str = greeting.substr();
```

greeting	H	e	l	l	o		W	o	r	l	d	!
index	0	1	2	3	4	5	6	7	8	9	10	11

What value would pos take?

○ npos

● 0

✓ Correct; this is a pretty clear violation of the policy, including using another person's account and impersonating another individual. ×

What value would `len` take?

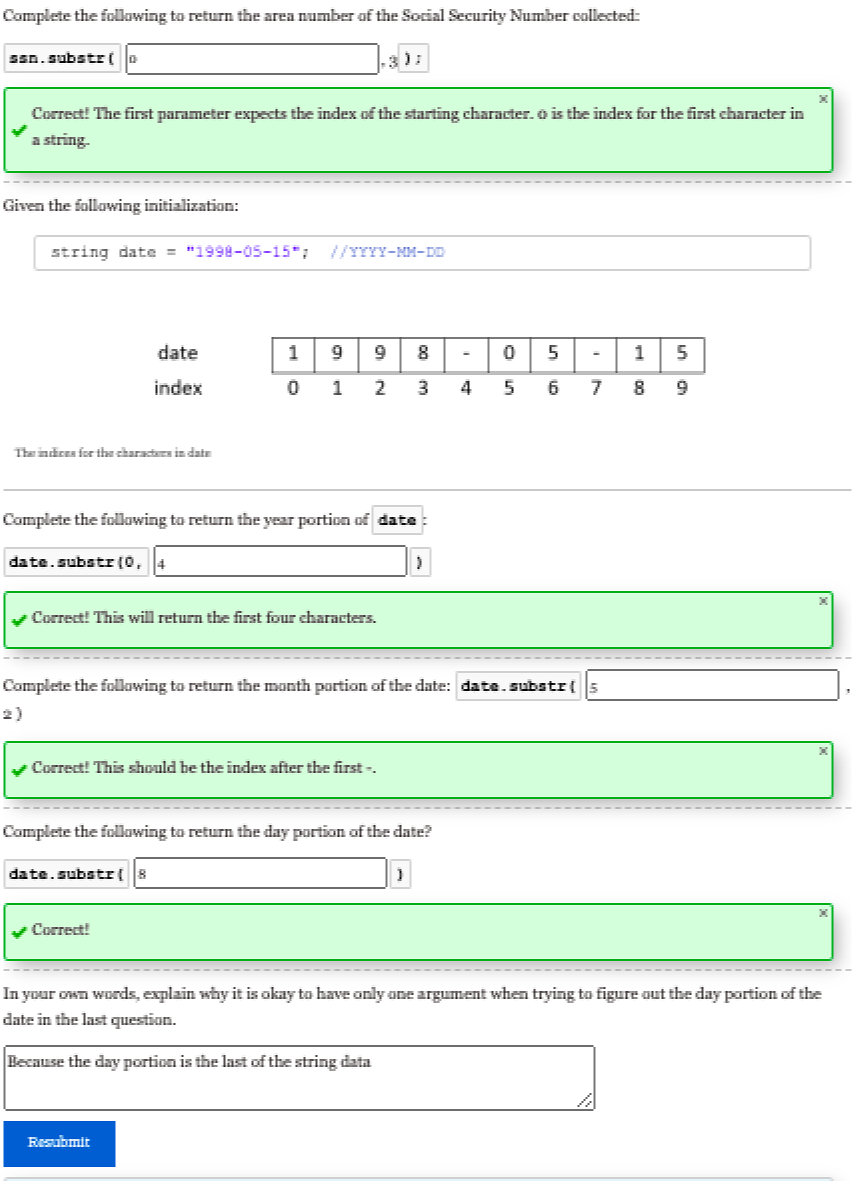
○ 0

● npos

✓ Correct. ×

What would be the value stored in str? "Hello World!"

✓ Correct. without any argument, the function call return the whole string of the invoking object. ×



Complete the following to return the area number of the Social Security Number collected:

`ssn.substr(` 0 , 3 `);`

✓ Correct! The first parameter expects the index of the starting character. 0 is the index for the first character in a string.

Given the following initialization:

```
string date = "1998-05-15";  //YYYY-MM-DD
```

date	1	9	9	8	-	0	5	-	1	5
index	0	1	2	3	4	5	6	7	8	9

The indices for the characters in date

Complete the following to return the year portion of `date`:

`date.substr(0,` 4 `)`

✓ Correct! This will return the first four characters.

Complete the following to return the month portion of the date: `date.substr(` 5 , 2)

✓ Correct! This should be the index after the first -.

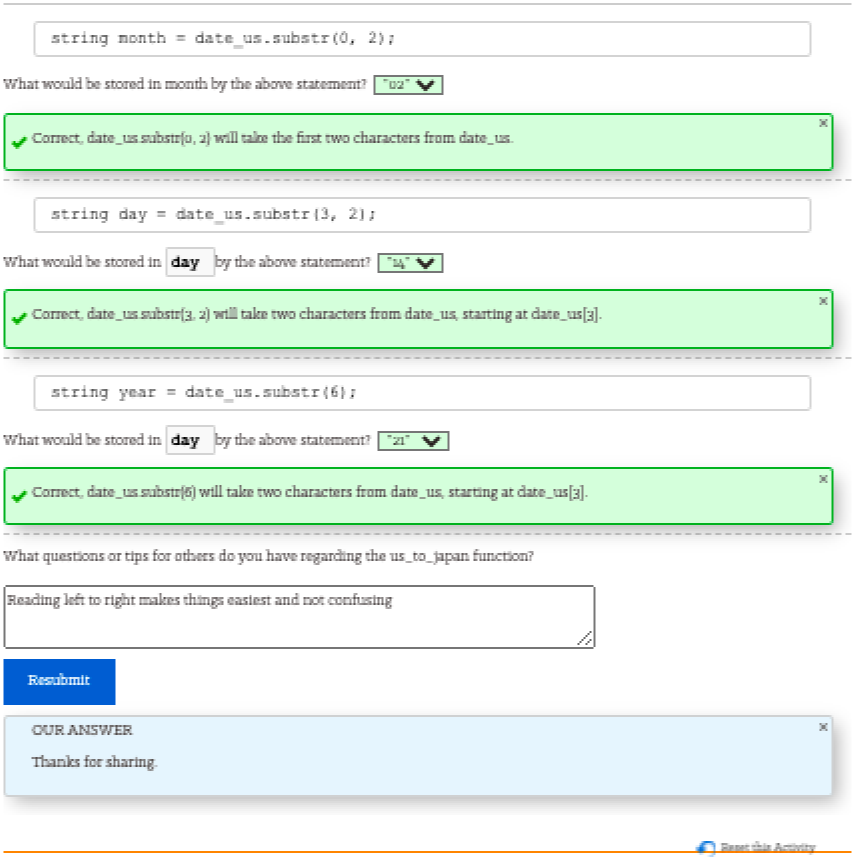
Complete the following to return the day portion of the date?

`date.substr(` 8 `)`

✓ Correct!

In your own words, explain why it is okay to have only one argument when trying to figure out the day portion of the date in the last question.

Because the day portion is the last of the string data



```
string month = date_us.substr(0, 2);
```

What would be stored in month by the above statement? "02"

✓ Correct, date_us.substr(0, 2) will take the first two characters from date_us.

```
string day = date_us.substr(3, 2);
```

What would be stored in `day` by the above statement? "14"

✓ Correct, date_us.substr(3, 2) will take two characters from date_us, starting at date_us[3].

```
string year = date_us.substr(6);
```

What would be stored in `day` by the above statement? "21"

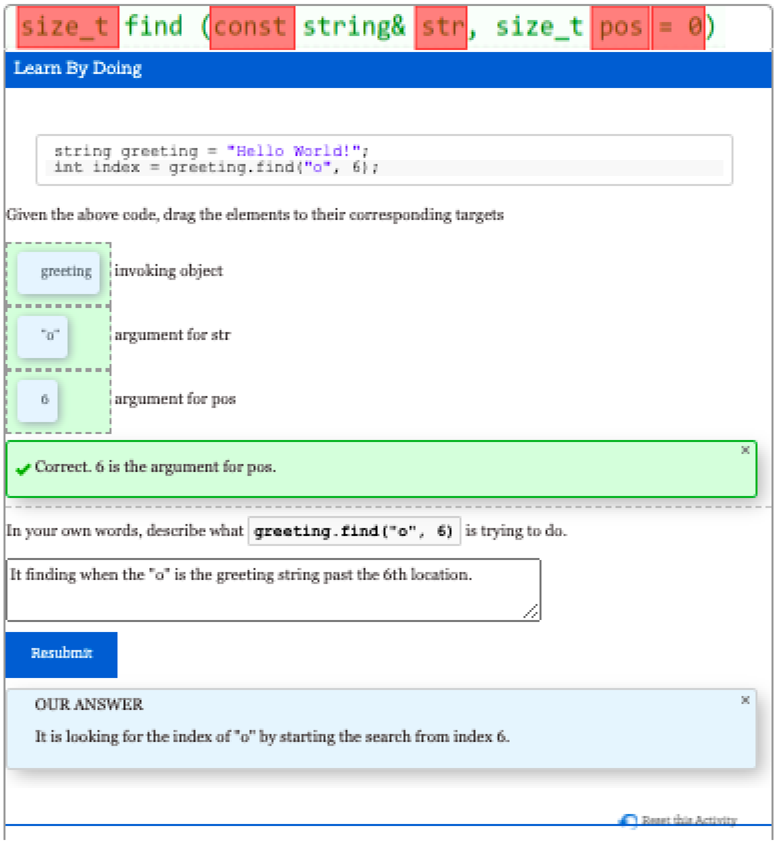
✓ Correct, date_us.substr(6) will take two characters from date_us, starting at date_us[3].

What questions or tips for others do you have regarding the us_to_japan function?

Reading left to right makes things easiest and not confusing

OUR ANSWER

Thanks for sharing.



```
size_t find (const string& str, size_t pos = 0)
```

Learn By Doing

```
string greeting = "Hello World!";
int index = greeting.find("o", 6);
```

Given the above code, drag the elements to their corresponding targets

greeting	invoking object
"o"	argument for str
6	argument for pos

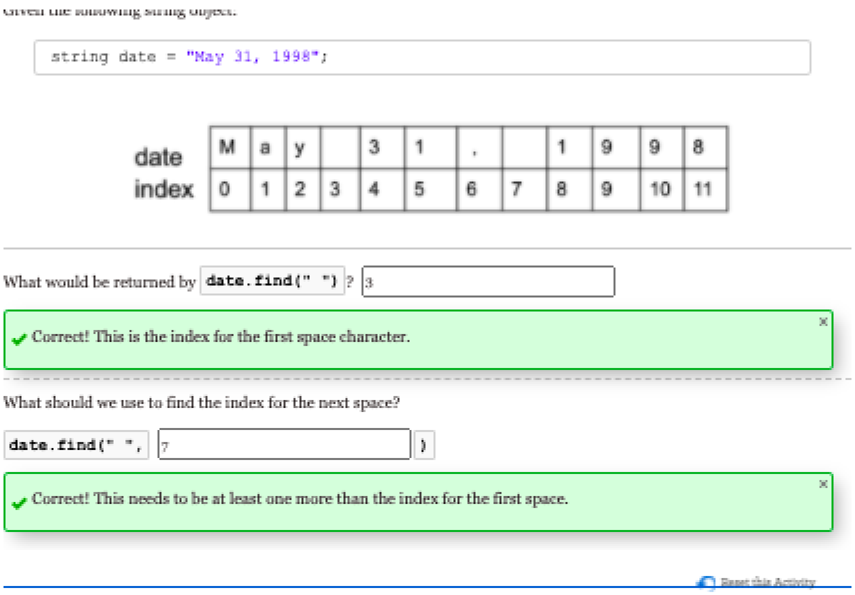
✓ Correct. 6 is the argument for pos.

In your own words, describe what `greeting.find("o", 6)` is trying to do.

It finding when the "o" is the greeting string past the 6th location.

OUR ANSWER

It is looking for the index of "o" by starting the search from index 6.



Given the following string object:

```
string date = "May 31, 1998";
```

date	M	a	y		3	1	,		1	9	9	8
index	0	1	2	3	4	5	6	7	8	9	10	11

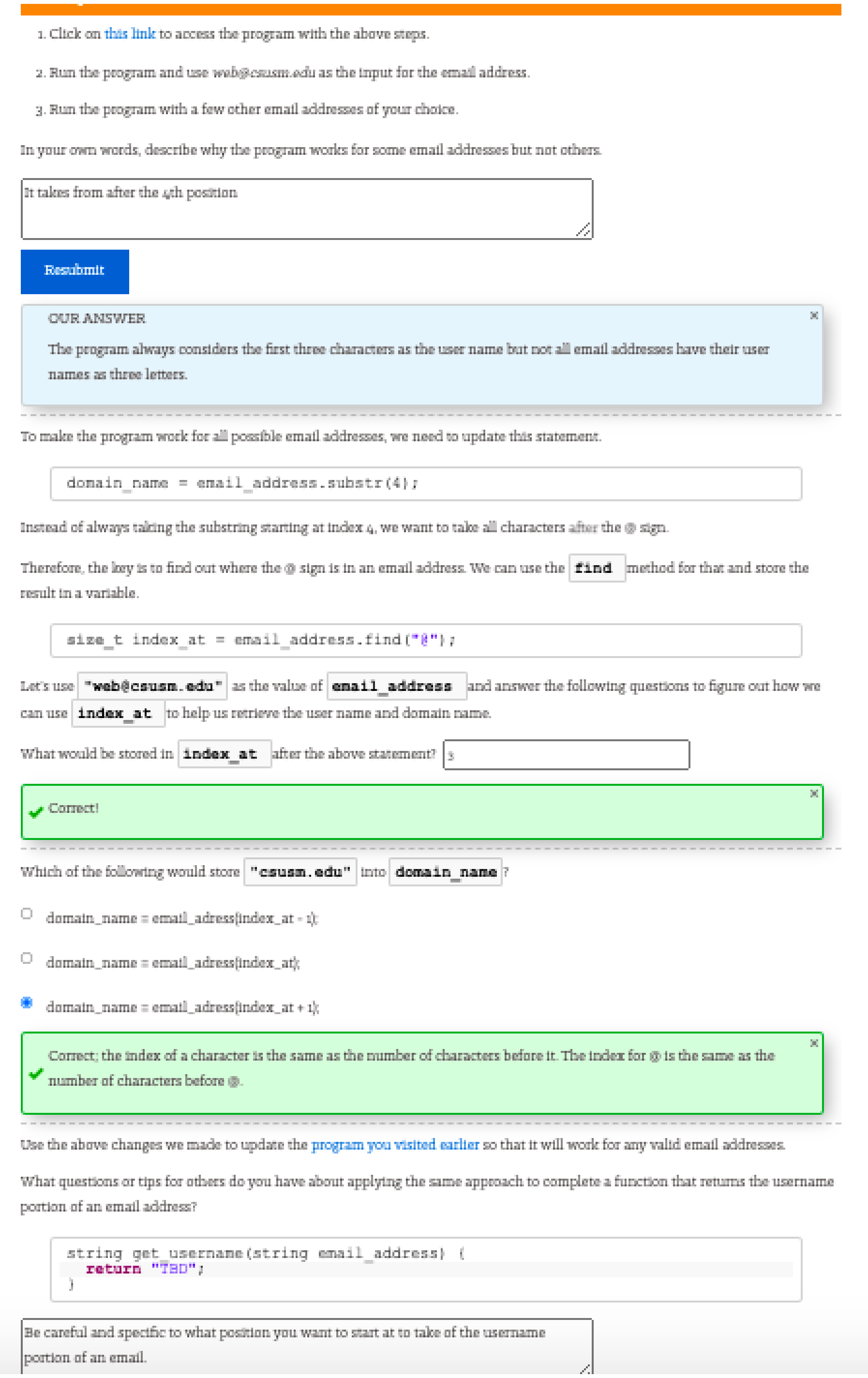
What would be returned by `date.find(" ")` ? 3

✓ Correct! This is the index for the first space character.

What should we use to find the index for the next space?

`date.find(" ",` 7 `)`

✓ Correct! This needs to be at least one more than the index for the first space.



1. Click on this link to access the program with the above steps.

2. Run the program and use *web@csusm.edu* as the input for the email address.

3. Run the program with a few other email addresses of your choice.

In your own words, describe why the program works for some email addresses but not others.

It takes from after the 4th position

OUR ANSWER

The program always considers the first three characters as the user name but not all email addresses have their user names as three letters.

To make the program work for all possible email addresses, we need to update this statement.

```
domain_name = email_address.substr(4);
```

Instead of always taking the substring starting at index 4, we want to take all characters after the @ sign.

Therefore, the key is to find out where the @ sign is in an email address. We can use the `find` method for that and store the result in a variable.

```
size_t index_at = email_address.find("@");
```

Let's use `"web@csusm.edu"` as the value of `email_address` and answer the following questions to figure out how we can use `index_at` to help us retrieve the user name and domain name.

What would be stored in `index_at` after the above statement? 3

✓ Correct!

Which of the following would store `"csusm.edu"` into `domain_name`?

○ domain_name = email_adress(index_at - 1);

○ domain_name = email_adress(index_at);

● domain_name = email_adress(index_at + 1);

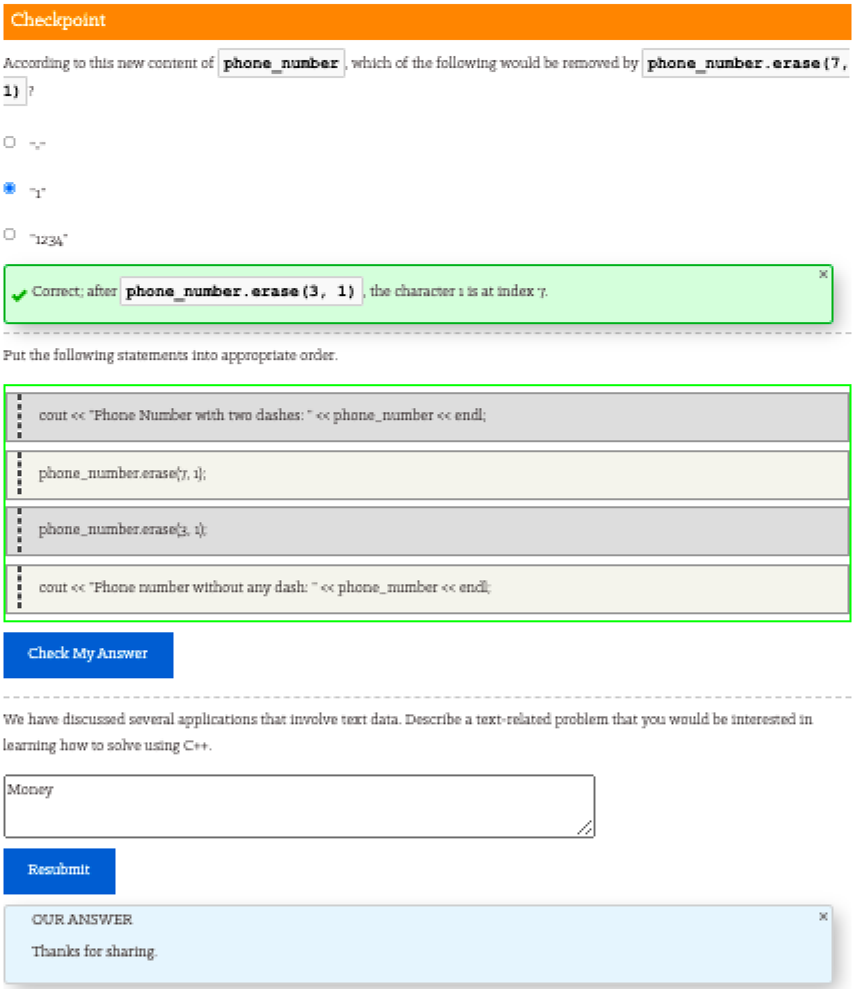
✓ Correct; the index of a character is the same as the number of characters before it. The index for @ is the same as the number of characters before @.

Use the above changes we made to update the program you visited earlier so that it will work for any valid email addresses.

What questions or tips for others do you have about applying the same approach to complete a function that returns the username portion of an email address?

```
string get_username(string email_address) {
  return "TBD";
}
```

Be careful and specific to what position you want to start at to take of the username portion of an email.



Checkpoint

According to this new content of `phone_number`, which of the following would be removed by `phone_number.erase(7, 1)`?

○ "-,-"

● "1"

○ "1234"

✓ Correct; after `phone_number.erase(3, 1)`, the character 1 is at index 7.

Put the following statements into appropriate order.

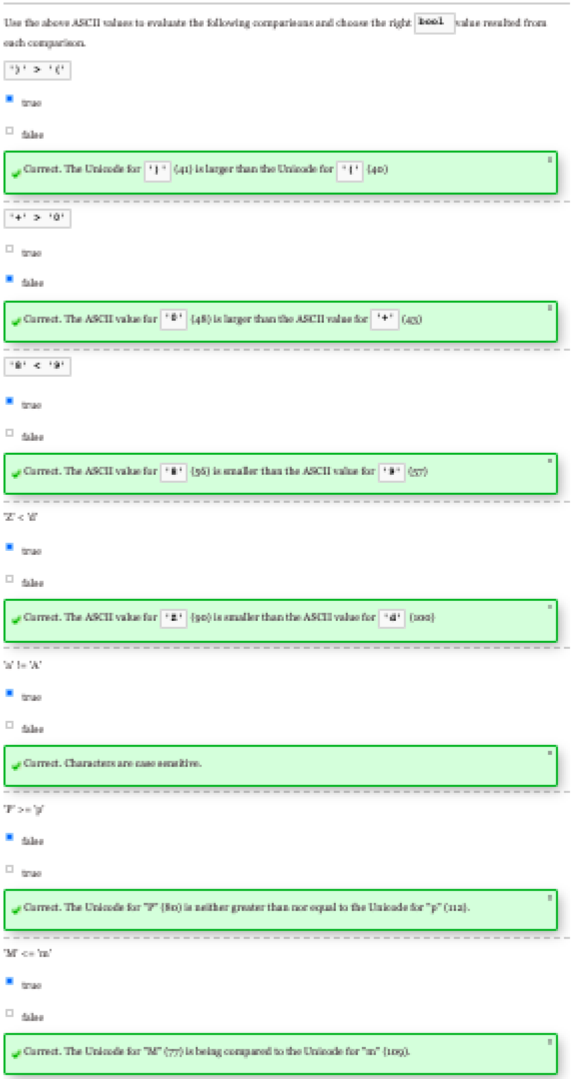
```
cout << "Phone Number with two dashes: " << phone_number << endl;
phone_number.erase(7, 1);
phone_number.erase(3, 1);
cout << "Phone number without any dash: " << phone_number << endl;
```

We have discussed several applications that involve text data. Describe a text-related problem that you would be interested in learning how to solve using C++.

Money

OUR ANSWER

Thanks for sharing.



Use the above ASCII values to evaluate the following comparisons and choose the right `bool` value resulted from each comparison.

`')' > '('`

☑ true

□ false

✓ Correct. The Unicode for `')'` (41) is larger than the Unicode for `'('` (40)

`'+' > '0'`

□ true

☑ false

✓ Correct. The ASCII value for `'0'` (48) is larger than the ASCII value for `'+'` (43)

`'8' < '9'`

☑ true

□ false

✓ Correct. The ASCII value for `'8'` (56) is smaller than the ASCII value for `'9'` (57)

'Z' < 'd'

☑ true

□ false

✓ Correct. The ASCII value for `'Z'` (90) is smaller than the ASCII value for `'d'` (100)

'a' != 'A'

☑ true

□ false

✓ Correct. Characters are case sensitive.

'P' >= 'p'

☑ false

□ true

✓ Correct. The Unicode for "P" (80) is neither greater than nor equal to the Unicode for "p" (112).

'M' <= 'm'

☑ true

□ false

✓ Correct. The Unicode for "M" (77) is being compared to the Unicode for "m" (109).

Comparing "abcd" with "aBcde"

Checking the strings from left to right, what is the first index where the characters differ?

1

The result of `"abcd" < "aBcde"` is false

✓ Correct! 'b' does not equal 'B'.

✓ Correct! Because `'b'` is greater than `'B'`

String #1:	c	o	m	p	u	t	e	r
String #2:	c	o	m	p				
Index:	0	1	2	3	4	5	6	7

Comparing "computer" with "comp"

Checking the strings from left to right, what is the first index where the characters differ?

4

The result of `"computer" < "comp"` is false

✓ Correct! String #2 ends after the 3rd index.

✓ Correct! "computer" is actually greater than "comp".

String #1:	C	S		2	1	1
String #2:	C	S	1	1	1	
Index:	0	1	2	3	4	5

Comparing "CS 211" with "CS111"

Checking the strings from left to right, what is the first index where the characters differ?

2

The result of `"CS 211" < "CS111"` is true

✓ Correct! Because `' '` does not equal `'1'`

✓ Correct! Because the space character has lower ASCII value than digits. That is `' '` is less than `'1'`

String #1:	C	o	u	g	a	r	s
String #2:	C	S	U	S	M		
Index:	0	1	2	3	4	5	6

Comparing "Cougars" with "CSUSM"

Checking the strings from left to right, what is the first index where the characters differ?

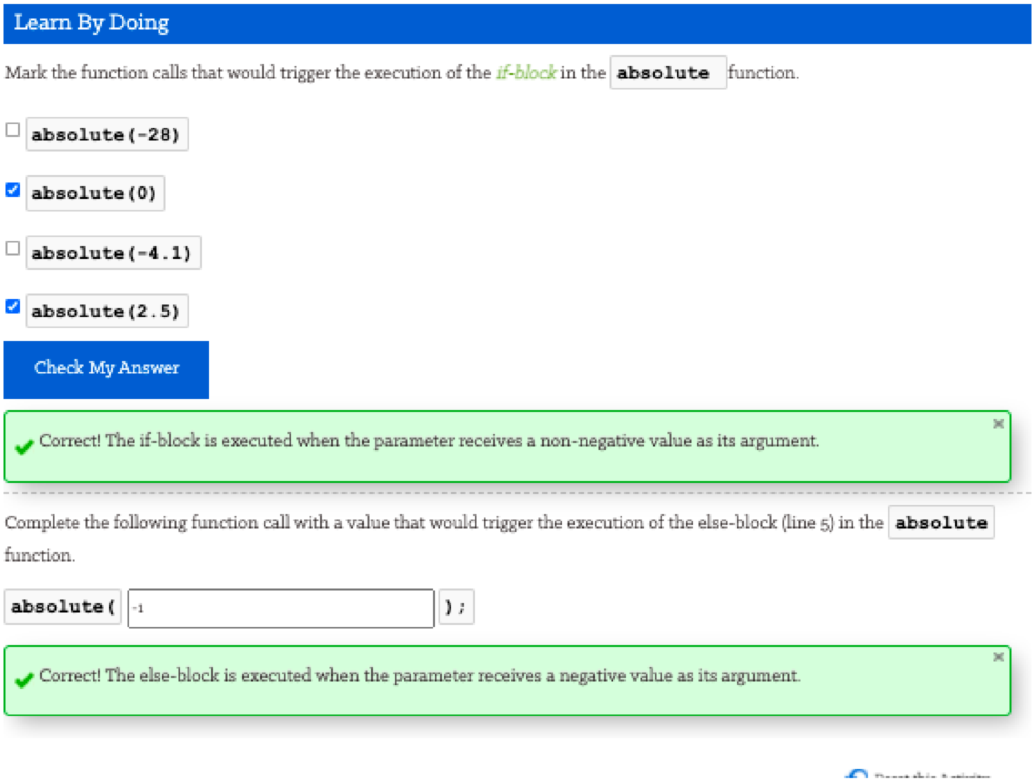
1

The result of `"Cougars" < "CSUSM"` is false

✓ Correct! Because `'o'` does not equal `'S'`

✓ Correct! Because `'o'` (ASCII value is 111) is greater than `'S'` (ASCII value is 83)

Report this Activity



Learn By Doing

Mark the function calls that would trigger the execution of the *if-block* in the `absolute` function.

☐ `absolute(-28)`

☑ `absolute(0)`

☐ `absolute(-4.1)`

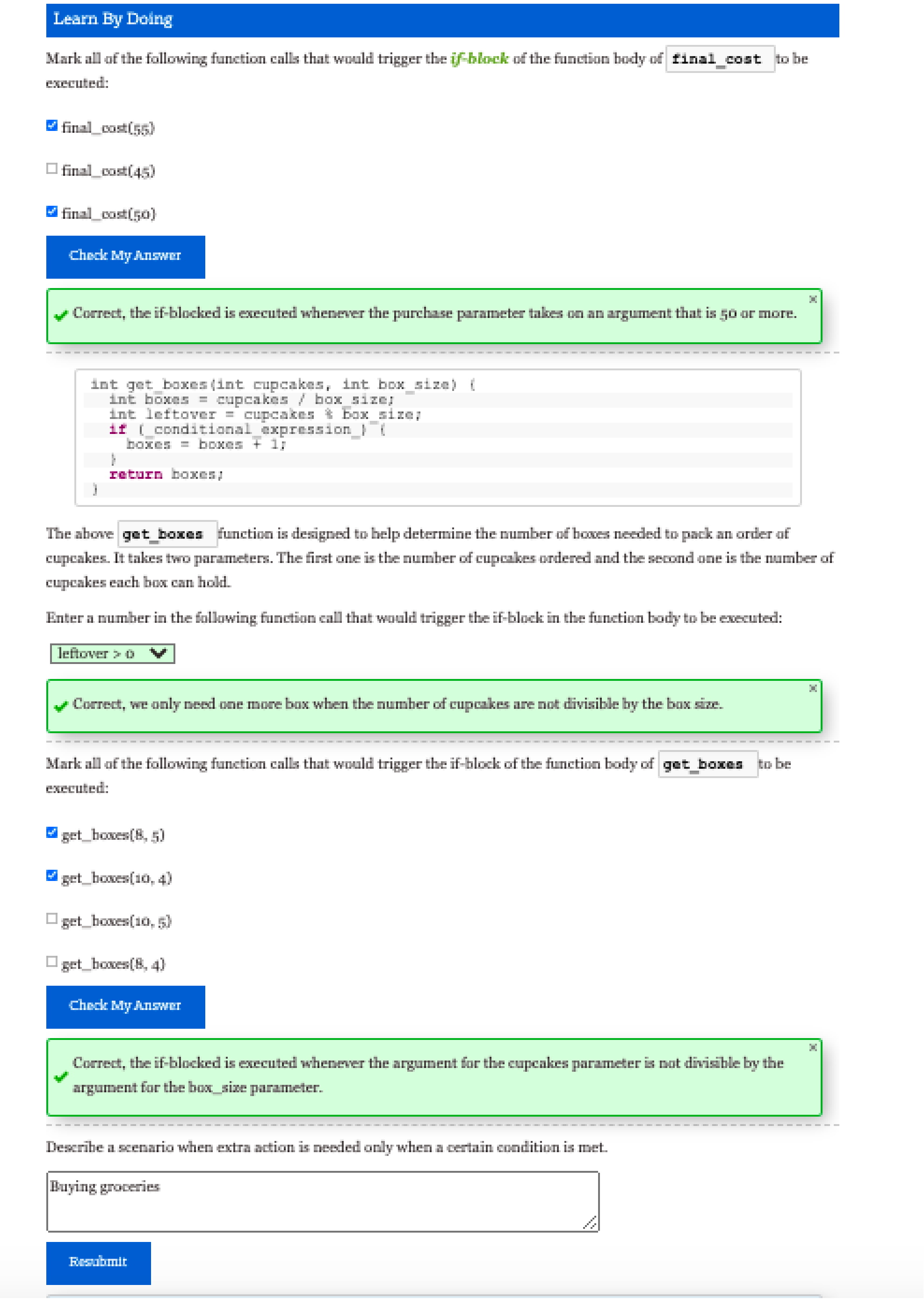
☑ `absolute(2.5)`

✓ 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.



Learn By Doing

Mark all of the following function calls that would trigger the *if-block* of the function body of `final_cost` to be executed:

☑ final_cost(55)

□ final_cost(45)

☑ final_cost(50)

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

```
int get_boxes(int cupcakes, int box_size) {
  int boxes = cupcakes / box_size;
  int leftover = cupcakes % box_size;
  if (_conditional_expression_) {
    boxes = boxes + 1;
  }
  return boxes;
}
```

The above `get_boxes` function is designed to help determine the number of boxes needed to pack an order of cupcakes. It takes two parameters. The first one is the number of cupcakes ordered and the second one is the number of cupcakes each box can hold.

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

leftover > 0

✓ Correct, we only need one more box when the number of cupcakes are not divisible by the box size.

Mark all of the following function calls that would trigger the if-block of the function body of `get_boxes` to be executed:

☑ get_boxes(8, 5)

☑ get_boxes(10, 4)

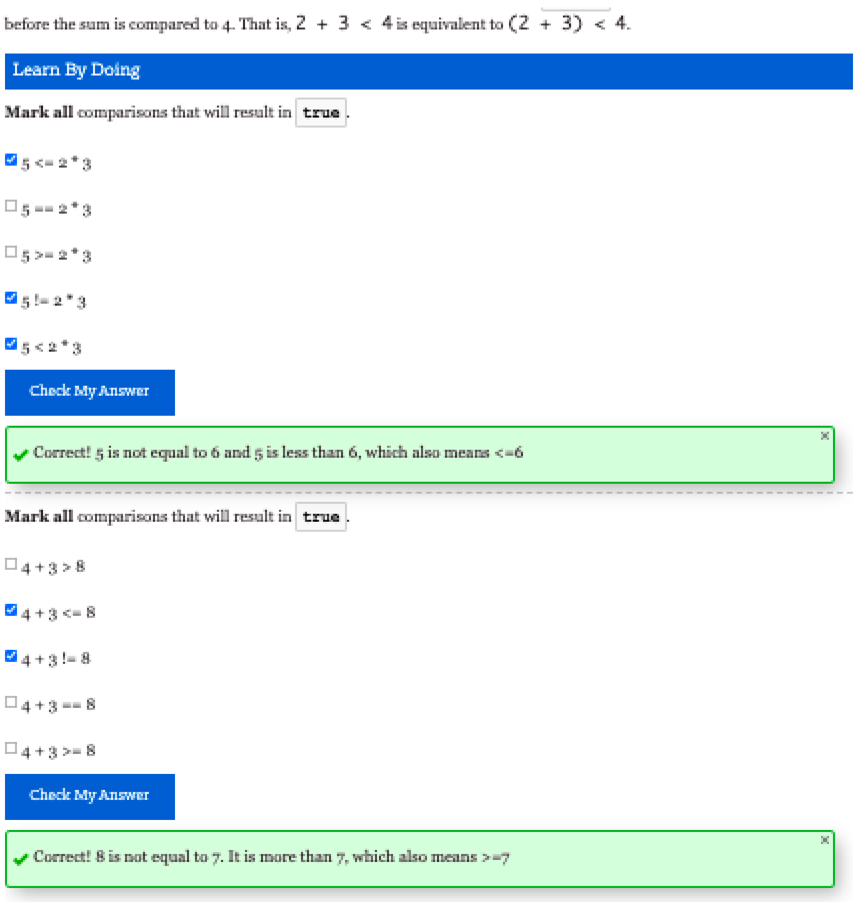
□ get_boxes(10, 5)

□ get_boxes(8, 4)

✓ Correct, the if-blocked is executed whenever the argument for the cupcakes parameter is not divisible by the argument for the box_size parameter.

Describe a scenario when extra action is needed only when a certain condition is met.

Buying groceries



before the sum is compared to 4. That is, `2 + 3 < 4` is equivalent to `(2 + 3) < 4`.

Learn By Doing

Mark all comparisons that will result in `true`.

☑ 5 <= 2 * 3

□ 5 == 2 * 3

□ 5 >= 2 * 3

☑ 5 != 2 * 3

☑ 5 < 2 * 3

Check My Answer

✓ Correct! 5 is not equal to 6 and 5 is less than 6, which also means <=6 ×

Mark all comparisons that will result in `true`.

□ 4 + 3 > 8

☑ 4 + 3 <= 8

☑ 4 + 3 != 8

□ 4 + 3 == 8

□ 4 + 3 >= 8

Check My Answer

✓ Correct! 8 is not equal to 7. It is more than 7, which also means >=7 ×