

Courses

78

Groups

Calendar

Inbox

0

Commons

?

Help

Spring 2020

Home

Announcements **Syllabus**

Modules

Files

Proctortrack People

Canvas Help

Grades

BlueJeans

My Media Media Gallery

Well-Being Connect

RC01 - Java

Due No due date Points 1 **Questions** 7 Available Jan 6 at 12am - Mar 14 at 11:59pm 2 months

Time Limit 30 Minutes **Allowed Attempts** Unlimited

Instructions

Reality checks are participation grades. You are not graded on correctness or full

General Instructions:

some form of an answer provided. You are encouraged to discuss the questions with your classmates. Any TA will be able to answer your questions from the reality checks. **Specific Instructions:** For these questions, imagine the following scenario:

completion. You are only graded upon earnest attempts. The reality check questions have

Last Attempt Details:

Kept Score: 0 out of 1

4 Attempts so far

Unlimited Attempts

Time:

Current

Score:

less than 1

0 out of 1 *

minute

* Some questions not yet graded

(\subseteq View Previous Attempts

DrugProvider is an interface requiring a void fillPrescription() method. Class **GroceryStore** has a **void shop()** method.

Class Publix is a subclass of GroceryStore implementing DrugProvider and overriding shop().

This quiz was locked Mar 14 at 11:59pm.

Attempt History Attempt Time

Attempt 4

KEPT

Correct!

LATEST	Attempt 4	less than 1 minute	0 out of 1 *	
	Attempt 3	less than 1 minute	0 out of 1 *	
	Attempt 2	less than 1 minute	0 out of 1 *	
	Attempt 1	18 minutes	0 out of 1*	
* Some questions not yet graded				
Score for this attempt: 0 out of 1 * Submitted Jan 6 at 10:29pm				

less than 1 minute

Score

Not yet graded / 1 pts

0 out of 1 *

This attempt took less than 1 minute. Question 1

When did you take Georgia Tech's CS 1331 (or explain your

Your Answer:

Fall 2019

transfer credit if that is the case (school/course))?

Question 2	0 / 0 pts			
For this question, imagine the following scenario: DrugProvider is an interface requiring a void fillPrescription() method.				
Class GroceryStore has a void shop() method.				

Class **Publix** is a subclass of **GroceryStore** implementing DrugProvider and overriding shop(). Which shop method is invoked from this code? (Choose one answer.) GroceryStore g = new Publix(); g.shop(); shop from Publix runs shop from GroceryStore runs onone - this code will not compile shop from DrugProvider runs

onone - this code will compile, but not run successfully

For this question, imagine the following scenario:

Question 3

0 / 0 pts

DrugProvider is an interface requiring a **void fillPrescription()** method. Class **GroceryStore** has a **void shop()** method. Class **Publix** is a subclass of **GroceryStore** implementing DrugProvider and overriding shop(). Which **toString** method is invoked from this code assuming Publix does not override the toString method? (Choose one answer.) DrugProvider d = new Publix(); System.out.println(d); • toString from nearest superclass implementing toString runs

onone - this code will compile, but not run successfully

none - this code will not compile

onone - toString is not called

toString from Publix runs

Correct!

Correct!

Correct!

Correct!

toString from Object runs 0 / 0 pts **Question 4** For this question, imagine the following scenario: **DrugProvider** is an interface requiring a **void fillPrescription()** method. Class GroceryStore has a void shop() method.

GroceryStore g2 = new GroceryStore(); ((DrugProvider)g2).fillPrescription(); fillPrescription from GroceryStore runs

fillPrescription from DrugProvider runs

fillPrescription from Publix runs

Class **Publix** is a subclass of **GroceryStore**

(Choose one answer.)

Question 5

method.

method.

implementing DrugProvider and overriding shop().

Which fillPrescription method is invoked from this code?

onone - this code will not compile • none - this code will compile, but not run successfully

DrugProvider is an interface requiring a **void fillPrescription()**

For this question, imagine the following scenario:

Class GroceryStore has a void shop() method.

implementing DrugProvider and overriding shop().

Class **Publix** is a subclass of **GroceryStore**

0 / 0 pts

Which **shop** method is invoked from this code? (Choose one answer.) Object o = new Publix(); ((GroceryStore)o).shop(); shop from Publix runs onone - this code will not compile shop from GroceryStore runs none - this code will compile, but not run successfully shop from DrugProvider runs 0 / 0 pts **Question 6**

For this question, imagine the following scenario:

Class GroceryStore has a void shop() method.

implementing DrugProvider and overriding shop().

Which **fillPrescription** method is invoked from this

Class **Publix** is a subclass of **GroceryStore**

DrugProvider is an interface requiring a void fillPrescription()

code? (Choose one answer.) Publix p = new GroceryStore(); ((DrugProvider)p).fillPrescription(); • none - this code will not compile fillPrescription from Publix runs fillPrescription from DrugProvider runs onone - this code will compile, but not run successfully fillPrescription from GroceryStore runs Not yet graded / 0 pts **Question 7** Assume you are adding a boolean-valued instance method called increasingOrder to a pre-existing class. The class has an array of java.lang.Comparable<T> as an instance field called data. Your instance method is to determine if the instance field array, data, is in increasing order. The method is to return true

otherwise. Assume there are no duplicates in the array. You may assume the array itself is not null, but it is possible to have null values in

> the array itself. If so, then the array is not in increasing order. Efficiency matters. Your solution should run in O(n) time, and at most a single passthrough should suffice. Also, you are not allowed to sort, nor make a copy of the array (O(1) space). Write only the instance method, not the class. Your Answer:

if the values in the array are found to be increasing order, false

public boolean increasingOrder() { $if(data.length > 0 \&\& data[0] == null) {$ return false;

```
for(int i = 0; i < data.length - 1; i++)
        if(data[i + 1] == null || data[i].compareTo(data
[i + 1]) > 0) {
             return false;
     return true;
}
   public boolean increasingOrder() {
        if(data.length > 0 \&\& data[0] == null) {
```

return false; for(int i = 0; i < data.length - 1; i++) if(data[i + 1] == null || data[i].compareT $o(data[i + 1]) > 0) {$ return false; return true;

Quiz Score: 0 out of 1

Next ▶

 \vdash