



The University of the West Indies, St. Augustine
COMP 2603 Object Oriented Programming I
Assignment 1 Grade Sheet
2020/2021 Semester 2

Hansel Charles (816011166)

86.63%

A

Criteria	Mark
VirtualMeetingSystem	18.0
VirtualRoom	23.0
BreakoutRoom	21.0
Participant	7.5
Bonuses	+5.0
Total (out of 86.0)	74.5

Grade Changes

Observe any deductions or bonuses that you have incurred or earned.

Bonuses	
Question Bonus: allocateParticipants(String)	+5.0

VirtualMeetingSystem Class

Passed 8/10; Partially passed 1/10; Failed 1/10.

Method	<code>createVirtualRoom(String)</code>	2.0 / 2.0
Method	<code>addParticipant(String, int)</code>	2.0 / 2.0
Method	<code>listParticipants(int)</code>	3.0 / 3.0
Method	<code>openBreakoutRoom(int)</code>	2.0 / 2.0
Method	<code>closeBreakoutRoom(int)</code>	2.0 / 2.0
Method	<code>listAllBreakoutRooms()</code>	2.0 / 2.0
Method	<code>findParticipantBreakoutRoom(String)</code>	2.0 / 2.0
Method	<code>listParticipantsInAllBreakoutRooms()</code>	2.0 / 2.0
Method	<code>allocateParticipants(String)</code>	0.0 / 10.0
Does not work as anticipated: produced java.lang.NullPointerException: Cannot invoke "String.length()" because "participantID" is null. We expect your method to run without problems, but instead yours contains bad code that creates problems.		
Check <code>VirtualMeetingSystem()</code> constructor exists and is accessible		✓
Check <code>loadParticipantData(String)</code> method exists and is accessible		✓
Check <code>createVirtualRoom(String)</code> method exists and is accessible		✓
Check <code>public void allocateParticipants(String)</code> method exists and defined properly		✓

Check <code>VirtualMeetingSystem()</code> constructor creates instances	✓
Check <code>loadParticipantData(String)</code> method runs with args (<code>"src/al/test/resources/participant.dat"</code>)	✓
Check <code>createVirtualRoom(String)</code> method runs with args (<code>"VirtualRoom"</code>)	✓
Check <code>allocateParticipants(String)</code> method runs with args (<code>"RR"</code>)	✓

Method	<code>loadParticipantData(String)</code>	1.0 / 5.0
Does not alter the data attribute as it should. We expect your method to change the values of a particular set of instance attributes, but yours doesn't, or it does it in an unanticipated way.		
Check <code>VirtualMeetingSystem()</code> constructor exists and is accessible		✓
Check <code>public void loadParticipantData(String)</code> method exists and defined properly		✓
Check <code>VirtualMeetingSystem()</code> constructor creates instances		✓
Check <code>loadParticipantData(String)</code> method runs with args (<code>"src/al/test/resources/participant.dat"</code>)		✓
Check first attribute with type <code>String[]</code> equals an array with size 50		✓
Check data attribute equals an array with size 50 +1.0		✓
Check first attribute with type <code>String[]</code> equals not an array containing null +1.0		✓
Check data attribute equals not an array containing null		✗

VirtualRoom Class

Passed 13/14; Failed 1/14.

Attribute	<code>breakoutRooms</code>	1.0 / 1.0
Attribute	<code>breakoutRoomLimit</code>	1.0 / 1.0
Attribute	<code>name</code>	1.0 / 1.0
Constructor	<code>VirtualRoom(String, int)</code>	2.0 / 2.0

Constructor	<code>VirtualRoom(String)</code>	2.0 / 2.0
Method	<code>listBreakoutRooms()</code>	2.0 / 2.0
Method	<code>findBreakoutRoom(int)</code>	2.0 / 2.0
Method	<code>createBreakoutRooms()</code>	2.0 / 2.0
Method	<code>openBreakoutRoom(int)</code>	2.0 / 2.0
Method	<code>closeBreakoutRoom(int)</code>	2.0 / 2.0
Method	<code>findParticipantBreakoutRoom(String)</code>	2.0 / 2.0
Method	<code>listParticipantsInBreakoutRoom(int)</code>	2.0 / 2.0
Method	<code>addParticipantToBreakoutRoom(String, int)</code>	2.0 / 2.0
Method	<code>getNumberOfBreakoutRooms()</code>	0.0 / 1.0
<p>Does not work as anticipated: produced java.lang.NullPointerException: Cannot invoke "Object.equals(Object)" because "this.breakoutRooms[i]" is null. We expect your method to run without problems, but instead yours contains bad code that creates problems.</p> <p>Check <code>VirtualRoom(String)</code> constructor exists and is accessible ✓</p> <p>Check <code>breakoutRoomLimit</code> attribute exists ✓</p> <p>Check <code>public int getNumberOfBreakoutRooms()</code> method exists and defined properly ✓</p> <p>Check <code>VirtualRoom(String)</code> constructor creates instances with args (<code>"VirtualRoom"</code>) ✓</p> <p>Check <code>getNumberOfBreakoutRooms()</code> method returns value equal to attribute <code>breakoutRoomLimit</code> ✗</p>		

BreakoutRoom Class

Passed 15/16; Partially passed 1/16.

Attribute	<code>breakoutRoomID</code>	1.0 / 1.0
-----------	-----------------------------	-----------

Attribute	<code>breakoutRoomSize</code>	1.0 / 1.0
Attribute	<code>participants</code>	1.0 / 1.0
Attribute	<code>numberOfParticipants</code>	1.0 / 1.0
Attribute	<code>open</code>	1.0 / 1.0
Attribute	<code>breakoutRoomNumberCounter</code>	1.0 / 1.0
Constructor	<code>BreakoutRoom(String)</code>	3.0 / 3.0
Method	<code>findParticipant(String)</code>	2.0 / 2.0
Method	<code>getBreakoutRoomID()</code>	1.0 / 1.0
Method	<code>getOpen()</code>	1.0 / 1.0
Method	<code>addParticipant(String)</code>	2.0 / 2.0
Method	<code>listParticipants()</code>	2.0 / 2.0
Method	<code>openBreakoutRoom()</code>	1.0 / 1.0
Method	<code>closeBreakoutRoom()</code>	1.0 / 1.0
Method	<code>getNumberOfParticipants()</code>	1.0 / 1.0
Method	<code>toString()</code>	1.0 / 2.0
<p>Returns correct information in an incorrect format. We expect your method to return a particular value in a given format, but instead yours returns the value in another format.</p>		
Check <code>BreakoutRoom(String)</code> constructor exists and is accessible		✓
Check <code>breakoutRoomID</code> attribute exists		✓
Check <code>numberOfParticipants</code> attribute exists		✓
Check <code>public String toString()</code> method exists and defined properly		✓
Check <code>BreakoutRoom(String)</code> constructor creates instances with args ("Room1")		✓

Check <code>openBreakoutRoom()</code> method runs	✓
Check <code>toString()</code> method returns string containing attribute <code>breakoutRoomID</code> +0.5	✓
Check <code>toString()</code> method returns string containing attribute <code>numberOfParticipants</code> +0.5	✓
Check <code>toString()</code> method returns string containing attribute <code>breakoutRoomID</code> in format <code>breakoutRoomID OPEN</code>	✗

Participant Class

Passed 4/5; Partially passed 1/5.

Attribute	<code>participantID</code>	1.0 / 1.0
Constructor	<code>Participant(String)</code>	2.0 / 2.0
Method	<code>verifyID(String)</code>	2.0 / 2.0
Method	<code>getParticipantID()</code>	1.0 / 1.0
Method	<code>toString()</code>	1.5 / 2.0

Returns correct information in an incorrect format. We expect your method to return a particular value in a given format, but instead yours returns the value in another format.

Check <code>Participant(String)</code> constructor exists and is accessible	✓
Check <code>participantID</code> attribute exists	✓
Check <code>public String toString()</code> method exists and defined properly	✓
Check <code>Participant(String)</code> constructor creates instances with args ("12345678")	✓
Check <code>toString()</code> method returns string containing attribute <code>participantID</code> +1.5	✓
Check <code>toString()</code> method returns string containing attribute <code>participantID</code> in format <code>Participant: participantID</code>	✗