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

Aaron Meighoo (816000303)

78.49%

A-

Criteria	Mark
VirtualMeetingSystem	15.0
VirtualRoom	24.0
BreakoutRoom	21.0
Participant	7.5
Total (out of 86.0)	**67.5**

VirtualMeetingSystem Class

Passed 7/10; Failed 3/10.

Method	**`createVirtualRoom(String)`**	2.0 / 2.0
Method	**`listParticipants(int)`**	3.0 / 3.0
Method	**`openBreakoutRoom(int)`**	2.0 / 2.0
Method	**`closeBreakoutRoom(int)`**	2.0 / 2.0
Method	**`listAllBreakoutRooms()`**	2.0 / 2.0
Method	**`findParticipantBreakoutRoom(String)`**	2.0 / 2.0
Method	**`listParticipantsInAllBreakoutRooms()`**	2.0 / 2.0
Method	**`loadParticipantData(String)`**	0.0 / 5.0

Does not work as anticipated: produced java.lang.NullPointerException: Cannot invoke "java.util.Scanner.hasNext()" because "this.readFromFile" is null. We expect your method to run without problems, but instead yours contains bad code that creates problems.

Check `VirtualMeetingSystem()` constructor exists and is accessible ✓

Check `public void loadParticipantData(String)` method exists and defined properly ✓

Check `VirtualMeetingSystem()` constructor creates instances ✓

Check `loadParticipantData(String)` method runs with args `("src/a1/test/resources/participant.dat")` ✗

Method	**`allocateParticipants(String)`**	0.0 / 10.0

Cannot be further tested because testing relies on the loadParticipantData method that does not work as anticipated: produced java.lang.NullPointerException: Cannot invoke "java.util.Scanner.hasNext()" because "this.readFromFile" is null. We expect your method to run without problems, but instead yours contains bad code that creates problems.

Check `VirtualMeetingSystem()` constructor exists and is accessible ✓

Check `loadParticipantData(String)` method exists and is accessible ✓

Check `createVirtualRoom(String)` method exists and is accessible ✓

Check `public void allocateParticipants(String)` method exists and defined properly ✓

Check `VirtualMeetingSystem()` constructor creates instances ✓

Check `loadParticipantData(String)` method runs with args `("src/a1/test/resources/participant.dat")` ✗

Method **`addParticipant(String, int)`** 0.0 / 2.0

Cannot be further tested because testing relies on the loadParticipantData method that does not work as anticipated: produced java.lang.NullPointerException: Cannot invoke "java.util.Scanner.hasNext()" because "this.readFromFile" is null. We expect your method to run without problems, but instead yours contains bad code that creates problems.

Check `VirtualMeetingSystem()` constructor exists and is accessible ✓

Check `loadParticipantData(String)` method exists and is accessible ✓

Check `createVirtualRoom(String)` method exists and is accessible ✓

Check `openBreakoutRoom(int)` method exists and is accessible ✓

Check `public boolean addParticipant(String, int)` method exists and defined properly ✓

Check `VirtualMeetingSystem()` constructor creates instances ✓

Check `loadParticipantData(String)` method runs with args `("src/a1/test/resources/participant.dat")` ✗

VirtualRoom Class

Passed 14/14.

Attribute **`breakoutRooms`** 1.0 / 1.0

Attribute **`breakoutRoomLimit`** 1.0 / 1.0

Attribute **`name`** 1.0 / 1.0

Constructor **`VirtualRoom(String, int)`** 2.0 / 2.0

Constructor	`VirtualRoom(String)`	2.0 / 2.0
Method	`listBreakoutRooms()`	2.0 / 2.0
Method	`findBreakoutRoom(int)`	2.0 / 2.0
Method	`createBreakoutRooms()`	2.0 / 2.0
Method	`openBreakoutRoom(int)`	2.0 / 2.0
Method	`closeBreakoutRoom(int)`	2.0 / 2.0
Method	`findParticipantBreakoutRoom(String)`	2.0 / 2.0
Method	`getNumberOfBreakoutRooms()`	1.0 / 1.0
Method	`listParticipantsInBreakoutRoom(int)`	2.0 / 2.0
Method	`addParticipantToBreakoutRoom(String, int)`	2.0 / 2.0

BreakoutRoom Class

Passed 15/16; Partially passed 1/16.

Attribute	`breakoutRoomID`	1.0 / 1.0
Attribute	`breakoutRoomSize`	1.0 / 1.0
Attribute	`participants`	1.0 / 1.0
Attribute	`numberOfParticipants`	1.0 / 1.0
Attribute	`open`	1.0 / 1.0
Attribute	`breakoutRoomNumberCounter`	1.0 / 1.0
Constructor	`BreakoutRoom(String)`	3.0 / 3.0
Method	`findParticipant(String)`	2.0 / 2.0

Method	`getBreakoutRoomID()`	1.0 / 1.0
Method	`getOpen()`	1.0 / 1.0
Method	`addParticipant(String)`	2.0 / 2.0
Method	`listParticipants()`	2.0 / 2.0
Method	`openBreakoutRoom()`	1.0 / 1.0
Method	`closeBreakoutRoom()`	1.0 / 1.0
Method	`getNumberOfParticipants()`	1.0 / 1.0
Method	`toString()`	1.0 / 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 `BreakoutRoom(String)` constructor exists and is accessible ✓

Check `breakoutRoomID` attribute exists ✓

Check `numberOfParticipants` attribute exists ✓

Check `public String toString()` method exists and defined properly ✓

Check `BreakoutRoom(String)` constructor creates instances with args `("Room1")` ✓

Check `openBreakoutRoom()` method runs ✓

Check `toString()` method returns string containing attribute `breakoutRoomID` +0.5 ✓

Check `toString()` method returns string containing attribute `numberOfParticipants` +0.5 ✓

Check `toString()` method returns string containing attribute `breakoutRoomID` in format `breakoutRoomID OPEN` ✗

Participant Class

Passed 4/5; Partially passed 1/5.

Attribute	**`participantID`**	1.0 / 1.0
Constructor	**`Participant(String)`**	2.0 / 2.0
Method	**`verifyID(String)`**	2.0 / 2.0
Method	**`getParticipantID()`**	1.0 / 1.0
Method	**`toString()`**	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 `Participant(String)` constructor exists and is accessible ✓

Check `participantID` attribute exists ✓

Check `public String toString()` method exists and defined properly ✓

Check `Participant(String)` constructor creates instances with args `("12345678")` ✓

Check `toString()` method returns string containing attribute `participantID` +1.5 ✓

Check `toString()` method returns string containing attribute `participantID` in format `Participant: participantID` ✗