

Lab Test 1 - Information

Lab Test 1 Venue

Conduct 2 sessions of lab (Test duration: 2hr)	Full-Time Students (Recess Week)		Part-Time Students
	2 Mar 2023 (Thursday)		2 Mar 2023 (Thurs)
	9.30 AM – 11.30 AM	13.30 PM – 15.30 PM	7.00 PM – 9.00 PM
SW1 – Rm Software Lab 1 (N4-01A-02)	B105	A112 A101	SW2 – Rm1 Software Lab 2 (N4-01C-06)
SW1 - Rm2 Software Lab 1 (N4-01A-02)	Z120 A120	B112	
SW2 – Rm1 Software Lab 2 (N4-01C-06)	A116 Z109	W117 A102	
SW2 – Rm2 Software Lab 2 (N4-01C-06)	A106 A119	Y116 B117	
HW1 – Rm1 Hardware Lab 1 (N4-01A-03)	C104 B120	B106 B116	
HW1 - Rm2 Hardware Lab 1 (N4-01A-03)	A104	Z107	
HW2 – Rm1 Hardware Lab 2 (N4-01B-05)	A115	C112	
HW2 - Rm2 Hardware Lab 2 (N4-01B-05)	B107	A108	
HW2 - Rm3 Hardware Lab 2 (N4-01B-05)	B115	B108	
HW2 – Rm4 Hardware Lab 2 (N4-01B-05)	B104	A107	
HPL – Rm1 Hardware Project Lab (N4-01C-09A)	C115	A117	
HPL – Rm1 Hardware Project Lab (N4-01C-09A)	A105	X117	

Lab Test 1 - Information

Instructions

1. This test comprises five (5) questions of 20 marks each
2. The questions could be from **Linked List, Stacks & Queues, Binary Trees, and Binary Search Trees.**
3. You are required to answer a total of **5 questions.**
4. You will have a total of **120 minutes** for this test.
5. System for the test: <https://www.hackerearth.com/>
6. **This is a closed-book test. You must not have any writing material, printed notes, or any electronic device with you other than the computer you are using to take the test. [Please note that you can only access <https://www.hackerearth.com/> and Code: Blocks during the test].**
7. During the test, you are not allowed to communicate with other people via any means.
8. Once the lab test starts, you cannot leave your seat until the lab test finishes.
9. Program templates for the Lab questions will be included inside the <https://www.hackerearth.com/> system. You must use them to implement your functions. The program contains a main () function, which consists of a switch statement to execute different functions that you should implement. This is similar to the three assignments. [Please note that all the utility functions will be provided within the program template.]
10. Please bring your matriculation card to the lab for the test.
11. Once the lab test is started, you must complete the test and follow all the instructions given by the lab executives, teaching assistants and lecturers.
12. **Please note that if you are absent in this test, you must submit the valid documents (e.g., MC) to support your absence by 02 March 2023 4:59PM to ofernando@ntu.edu.sg. Otherwise, you will receive 0 mark for this test. Late submission will not be entertained.**
13. Put your electronic device (such as hand phone) inside your bag and leave them on the floor. Do not use any of your mobile electronic devices during the lab session.
14. Rough paper will be provided. Please leave the rough paper on your desk after the test. You are not allowed to bring the rough paper with you after the test.

Lab Test 1 - Information

Questions Format for the Lab Test1

Question No	Lecture Moule	Descriptions
1	Linked List	From Labs tutorials, practice questions, and assignments 1. insertSortedLL() 2. alternateMergeLL() 3. moveoddItemsToBackLL() 4. frontBackSplitLL() 5. moveEvenItemsToBackLL() 6. moveMaxToFrontLL() 7. removeDuplicatesSortedLL() 8. split() 9. duplicateReverse() 10. concatenate() 11. combineAlternating()
2	Linked List	Unknown Question
3	Stacks and Queues	From Labs tutorials, practice questions, and assignments 1. createQueueFromLinkedList() 2. createStackFromLinkedList() 3. isStackPairwiseConsecutive() 4. reverseQueue() 5. removeUntil() 6. recursiveReverse() 7. palindrom()

Lab Test 1 - Information

		8. balance() 9. reverseStack() 10. reverseFirstKItems() 11. sortStack()
4	Stacks and Queues	Unknown Question
5	Binary Trees and Binary Search Trees	From Labs tutorials, practice questions, and assignments Please note that detailed algorithms are not provided for the lab test 1 1. identical() 2. maxHeight() 3. maxDepth() 4. countOneChildNodes() 5. sumOfOddNodes() 6. mirrorTree() 7. printSmallestValues() 8. smallestValue() 9. hasGreatGrandchild() 10. levelOrderTraversal() 11. inOrderTraversal() 12. preOrderIterative() 13. postOrderIterativeS1() 14. postOrderIterativeS2() 15. printBSTInOrder() 16. isBST()

Lab Test 1 - Information

Hackerearth (www.hackerearth.com) Instructions

1. Program templates for the Lab questions will be included inside the <https://www.hackerearth.com/> system. You must use them to implement your functions. The program contains a main () function consisting of a switch statement to execute different functions you should implement. This is like the three assignments. [Please note that all the utility functions will be provided within the program template.]
 1. After writing the code, you may (1) Compile and Test the Code; (2) Submit the code.
 2. When finished coding on all the questions, you may then “**End Test.**”
2. Please make sure that you submit your code successfully to the Hackerearth (www.hackerearth.com) System before you leave.
3. You can submit multiple times, but system will take the first **BEST attempt for the grading** [**The first BEST attempt: Assume that you got 90 on the first attempt. If you got 100 in the next attempt, which is better than 90, then the best attempt is 100 instead of 90. However, if you have already reached 100, there is no point in doing it again because you have already reached 100.**]
4. All the **necessary test cases are included for each question. Please note that the same test cases will use for the evaluation.**
5. Please note that **we do not entertain reset exercise during the Lab test**
6. Marking will be done automatically using the same test cases included in each question. Marks will be awarded according to the correctness of each test case.
7. Your submitted source code will be marked by an automatic C code marking component of the HackerEarth System according to the correctness of your program.
8. If **the question code cannot compile, you will get a 0 mark for that question.** Therefore, writing programs that can be compiled is essential.