SOFTWARE TESTING LABORATORY
(Effective from the academic year 2018 -2019)
SEMESTER – VI

Course Code	**18ISL66**	**CIE Marks**	40
Number of Contact Hours/Week	0:2:2	**SEE Marks**	60
Total Number of Lab Contact Hours	36	**Exam Hours**	03

Credits – 2

Course Learning Objectives: This course (18ISL66) will enable students to:

- Analyse the requirements for the given problem statement
- Design and implement various solutions for the given problem
- Employ various design strategies for problem solving.
- Construct control flow graphs for the solution that is implemented
- Create appropriate document for the software artefact

Descriptions (if any):

Design, develop, and implement the specified algorithms for the following problems using any language of your choice under LINUX /Windows environment.

Programs List:

1.	Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on boundary-value analysis, execute the test cases and discuss the results.
2.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.
3.	Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.
4.	Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on equivalence class partitioning, execute the test cases and discuss the results.
5.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of equivalence class testing, derive different test cases, execute these test cases and discuss the test results.
6.	Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of equivalence class value testing, derive different test cases, execute these test cases and discuss the test results.
7.	Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Derive test cases for your program based on decision-table approach, execute the test cases and discuss the results.
8.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of decision table-based testing, derive different test cases, execute these test cases and discuss the test results.
9.	Design, develop, code and run the program in any suitable language to solve the commission

	problem. Analyze it from the perspective of dataflow testing, derive different test cases, execute these test cases and discuss the test results.
10.	Design, develop, code and run the program in any suitable language to implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.
11.	Design, develop, code and run the program in any suitable language to implement the quicksort algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.
12.	Design, develop, code and run the program in any suitable language to implement an absolute letter grading procedure, making suitable assumptions. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results

Laboratory Outcomes: The student should be able to:

- List out the requirements for the given problem
- Design and implement the solution for given problem in any programming language(C,C++,JAVA)
- Derive test cases for any given problem
- Apply the appropriate technique for the design of flow graph.
- Create appropriate document for the software artefact.

Conduct of Practical Examination:

- All laboratory experiments, excluding the first, are to be included for practical examination.
- Experiment distribution
    - For questions having only one part: Students are allowed to pick one experiment from the lot and are given equal opportunity.
    - For questions having part A and B: Students are allowed to pick one experiment from part A and one experiment from part B and are given equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure part to be made zero.
- Marks Distribution *(Courseed to change in accoradance with university regulations)*

    m) For questions having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks

    n) For questions having part A and B

    i. Part A – Procedure + Execution + Viva = 4 + 21 + 5 = 30 Marks

    ii. Part B – Procedure + Execution + Viva = 10 + 49+ 11 = 70 Marks