**Computer Programming Lab**

This document contains the functional test cases of Computer Programming lab. It includes testing of internal and external links availability, also the expected output with valid inputs.

No. of Experiments: 10

Build requirements of lab: ubuntu-12, apache 2

**Test Environment:**

Operating Systems: Windows, CentOs, Ubuntu

Browsers: Chrome, Firefox and Midori

Landing page: All the links given on landing page of lab are working fine.

**Experiment wise Test Cases**

1. **Functions [1st Experiment]**

**Test Summary**:

To understand that a big program can be broken up into independent modules 2. To learn to define functions and call them with appropriate parameters

**Test Procedure:**

Click on procedure tab.

Follow the instructions given to perform the experiment by giving valid input.

Simulation should give the expected output to pass the test case.

**Test Data:**

<https://cse02-iiith.vlabs.ac.in/exp2/index.html>

**Links Availability:**

Broken link found:

Further readings Section: [www.cprogramming.com/tutorial/lesson16.html](https://cse02-iiith.vlabs.ac.in/exp2/www.cprogramming.com/tutorial/lesson16.html)

All other links are working fine.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

Click on any shape given in simulation to define a function for calculating the area of that shape.

Square:

1. Click on square.
2. Enter number of arguments: 1 (argument is a)
3. Enter datatype of arguments: float
4. Enter return datatype of the function: float
5. Choose formula for area of the square: a\*a

**Expected function to generate by simulation:**

//function for square

float area\_sq (float a)

{

  float area = a\*a;

  return area;

}

**Need to call appropriate function to find area with input:**

area\_sq(4); [Tested with many inputs]

**Expected output:**

16

**Test pass or Fail?**

**PASS**

**Comments:**

1. Instructions given in simulation are not clear. Needs to define properly.

**2. Numerical Approximation:**

**Test Summary**:

1. To understand that for some problems an approximate numerical solution is as good as the exact solution for practical purposes.

2. To understand how to apply these solutions to obtain arbitrary precision.

**Test Procedure:**

Click on procedure tab.

Follow the instructions given to perform the experiment by giving valid input.

Simulation should give the expected output to pass the test case.

**Test Data:**

<https://cse02-iiith.vlabs.ac.in/exp1/index.html>

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

1. Given a well-behaved smooth function, integrate it over the given limits of integration ?

2. Approximating solution to integral a smooth function.

3. Integration Limits are from 0 to 30, b > a and b-a >= 1.

**Sample input to test simulation:**

a=4.5, b=16;

**Procedure:**

After giving input, user needs to click on start button then next button till the end point.

**Expected output:**

LOCAL VARIABLES :  i = 16  sum = -0.12

OUTPUT :  INTEGRATION VALUE = -0.12

Along with graph from 4.5 to 16.

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

**3. Advanced Control Flow**

**Test Summary**:

1. To learn how decision making is done while programming.

2. To learn about the various advanced constructs used for control flow in order to achieve repetition of instructions.

**Test Procedure:**

Click on procedure tab.

Follow the instructions given to perform the experiment by giving valid input.

Simulation should give the expected output to pass the test case.

**Test Data:**

<https://cse02-iiith.vlabs.ac.in/exp3/index.html>

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

Choose experiment type by pressing the "Switch experiment" button on the bottom left.

Experiment 1:

1. Select the looping construct with which you want to solve the problem.
2. Enter the number for which factorial has to be calculated and press OK.
3. Press Start to start the experiment.
4. Click Next to get a step by step execution of the code.

Experiment 2:

1. Enter the height of pyramid and press OK.
2. Press Start to start the experiment.
3. Click Next to get a step by step execution of the code.

**Sample input to test simulation:**

Exp1: (Simple loop) for loop, While loop, do-while with value 5

Exp2: (Nested loop) height of pyramid = 5.

**Function**:

Exp1: fact=fact\*i;

**Procedure:**

After giving input, user needs to click on start button then next button till the end point.

To switch to experiment click on “switch to nested loop” button.

**Expected output:**

Exp 1: Simple loop

1 \* 5= 5

5 \* 4= 20

20 \* 3= 60

60 \* 2= 120

gives value of i and j along with highlighting each active line in c-program.

Exp2: nested loop

\_\_\_\_\*\_\_\_\_

\_\_\_\*\_\*\_\_\_

\_\_\*\_\*\_\*\_\_

\_\*\_\*\_\*\_\*\_

\*\_\*\_\*\_\*\_\*

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

**5. Arrays**

**Test Summary**:

1. To learn how to use arrays for storing large amount of data.

2. To learn how to use array to create a common reference for a large number of variables.

**Test Procedure:**

Click on procedure tab.

Follow the instructions given to perform the experiment by giving valid input.

Simulation should give the expected output to pass the test case.

**Test Data:**

<https://cse02-iiith.vlabs.ac.in/exp4/index.html>

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

1. Generate Random array
2. Enter values manually

**Sample input to test simulation:**

1D Array: Enter values manually: Entered array size: 4

Array values: 1,7,6,2,0

2D Array: Enter matrix size: 2 \* 2

**Procedure:**

After giving input, user needs to click on start button then next button till the end point.

**Expected output:**

1D Array: 0,1,2,6,7

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

6. **Basic Control Flow**

**Test Summary**:

1. To learn how decision making is done while programming.

2. To learn about the various simple constructs used for control flow.

**Test Procedure:**

**Experiment Part I: Position of Point w.r.t to a Rectangle**

*Here we shall see how the problem of determining whether a given point is inside or outside a rectangle can be solved using various conditional constructs.*

1. Initialize the values of the variables on the top left. X and Y denote the x and y coordinate of the input point.

2. Select the code prototype with which you would like to solve the problem.

3. Click start to begin the expriment.

4. Click Next to get a step by step execution of the code along with the reasoning which is displayed on the right hand side panel.

**Experiment Part II: Selecting the Day of a week**

*Here we shall see how a switching construct works by associating one number to each of the days.*

1. First, select a day from the top left by clicking on the correspoding radio button or enter any number in the textbox directly. Then, click on SUBMIT.

2. Select the code prototype with which you would like to solve the problem.

3. Click on the activated START button to begin the experiment.

4. Click Next to get a step by step execution of the code. The changes in local variables can be visualized on the bottom left of the screen. The OUTPUT can be visualized on the right part of the screen.

**Test Data:**

[https://cse02-iiith.vlabs.ac.in/exp6/index.html](https://cse02-iiith.vlabs.ac.in/exp2/index.html)

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

Test\_input: if-else code, if-else-if, X=150, Y=150

**Expected output:**

Local Variable :

flag\_1 = 1    flag\_2 = 1

flag\_3 = 1    flag\_4 = 1

x = 150      y = 150

if-else-if code: flag=1

Final point “outside INSIDE”

Experiment 2: Switch

Test input = 1 i.e monday

Expected output = Working Day

Actual Output = Working Day

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

**7. Strings**

**Test Summary**:

1. To understand the concept of strings and how they are a special type of character arrays.

2.To understand the usage of string libraries to do common string operations.

**Test Procedure:**

**String Matching**

1. *Press start to start the experiment and select two string str1 and str2 to compare.*
2. *Press next to see the execution of the code*
3. *Relevant line in the code is shown here*
4. *The output of the code is shown in the right*

**String Comparison**

1. *Press start to start the experiment and select two string str1 and str2 to compare.*
2. *Press next to see the execution of the code*
3. *The output of the code is shown in the right*
4. *You can stop the code using stop button*

**Test Data:**

[https://cse02-iiith.vlabs.ac.in/exp7/index.html](https://cse02-iiith.vlabs.ac.in/exp2/index.html)

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

Experiment 1: First String: Testing, Second String: Test

Experiment 2: First String: hello, Second String: bye

**Expected output:**

Experiment 1: Second string found in string 2.

Experiment 2: String1 is greater than string2.

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

**8. Strings**

**Test Summary**:

1. To understand the concept of strings and how they are a special type of character arrays.

2.To understand the usage of string libraries to do common string operations.

**Test Procedure:**

**Call By Value**

1. Press start to start the experiment.
2. Press next to see the execution of the code.
3. Relavant line in the code is shown here.
4. The output of the code is shown in the right.

**Call By Reference**

1. Press start to start the experiment.
2. Press next to see the execution of the code.
3. The output of the code is shown in the right.
4. You can stop the code using stop button.

**Test Data:**

[https://cse02-iiith.vlabs.ac.in/exp8/index.html](https://cse02-iiith.vlabs.ac.in/exp2/index.html)

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

Experiment 1: Pressed start button, then next button with default value of A = 10;

Experiment 2: Pressed start button, then next button, with default value of A=5 and B=9;

**Expected output:**

Experiment 1: New Value of A is 20

Experiment 2:

Value of A after swapping is 9.

Value of B after swapping is 5.

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

**9. Recursion**

**Test Summary**:

1. To understand that some problems can be broken down into smaller similar problems.
2. To solve such problems using recursive procedures.

**Test Procedure:**

1. Select the value of N (values must me greater than 1 and less than or equal to 5).
2. Press next to see the execution of the code.
3. Relevant line in the code is shown here.
4. The output of the code is shown in the right.

**Test Data:**

[https://cse02-iiith.vlabs.ac.in/exp9/index.html](https://cse02-iiith.vlabs.ac.in/exp2/index.html)

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

Number of disks: 4

**Expected output:**

Disks should places on one tower according to their size. biggest at the bottom, smallest on the top.

**Test pass or Fail?**

**PASS**

**Comments:**

No Comments

**10. Recursion**

**Test Summary**:

1. To learn about different types of operators.
2. To learn about the precedence of the operators.
3. To learn how to evaluate an expression.

**Test Procedure:**

Procedure for the experiment is as follows.

1. Select the type of operators and the datatype you want to work upon from the top most bar.
2. You can edit the values of variables by pressing the edit button.
3. Select an expression prototype from the menu.
4. You can also edit this expression.
5. You can also edit this expression.
6. Press Next to see the step by step evaluation of the selected expression in the central panel and the corresponding reasoning in the right panel.
7. Press stop if you want to abort the experiment and start over.

**Test Data:**

[https://cse02-iiith.vlabs.ac.in/exp10/index.html](https://cse02-iiith.vlabs.ac.in/exp2/index.html)

**Links Availability:**

All other links are working fine. Not a single broken link found.

**Procedure**

Procedure is to follow the “instructions” given in the lab to run “simulations” by giving valid input to get the expected output.

**Input**

a=2, b=5, c=10, d=11

1. Expression: a+b-c
2. (a+b)/c
3. (a+b)\*(c-d)

**Expected output:**

1. -3
2. 0.7
3. -7

**Test pass or Fail?**

**PASS**

**Comments:**

UI for this experiment does not load properly in firefox.