# Question No 1

# PART 1: 3 Marks

**Write down a program to check whether a statement is logically equivalent or not using truth table.**

**Using C++ and built-in libraries but you must have complete knowledge of it.**

# • Code must be commented accordingly otherwise marks will be deducted.

# • Attach the output in word document.

# • You have to display the truth table stepwise as follow.



# • You can take the following statements as an example but it should be generic.

**[(p ∧ q) → r] ↔ (p → (q → r)) ≡ [(p → q) ∧ (q → r)] → (p → r)**

# Question No 1

# PART 2: 3 Marks

**Write down a program to check whether an argument is valid or not using truth table. If it is valid then print the truth table, if not then highlight the specific row(s) in the truth table due to which the argument is invalid.**

**Using C++ and built-in libraries but you must have complete knowledge of it.**

# • Code must be commented accordingly otherwise marks will be deducted.

# • Attach the output in word document.

# • You have to display the truth table stepwise as in previous part.

# • You can take the following statements as an example but it should be generic.

**Premises: (p ∧ t) → (r ∨ s), q → (u ∧ t), u → p, and ¬s**

**Conclusion: q → r**

# Question No 2 4 Marks

**Write down a program to check the given graph whether it is Bipartite or not.**

**If not, then display the vertices which are causing conflict.**

**Using C++ and built-in libraries but you must have complete knowledge of it.**

# • Code must be commented accordingly otherwise marks will be deducted.

# • Attach the output in word document.

# • You have to implement it using adjacency matrix but it should be generic.

