

Logical Statements

Assignment:

Create a program that can try out every possible logical combination of the variables A, B, and C, and determine which combinations will yield a true statement. A few things to take note of, first there are eight different possible combinations of the three variables. Make certain you test all eight of the possible combinations of A, B, and C. (50 pts. for the code, 50 pts. for the screenshots)

- (1) (A and B) or (A and C)
- (2) (A and C) and (B and !C)
- (3) (A or B) and !(B or C)
- (4) (A or (B and C)) and (!A and !B)
- (5) ((B and C) or (C and A)) and !(A or B) and C

Please read each of the five carefully and make sure your code accurately reproduces the logical statement given for each problem. Note that the exclamation symbol indicates the logical "NOT" operator, and the parentheses indicate which order the statements are to be performed in. Keep in mind since you're testing all possible combinations of input for all five logical statements, you'll have FORTY answers generated by your program, in total.

Code:

```
// Name: Dawlat Hamad
// ID: GV5450
// Lab 2 - Truth Table
// Source 1 - https://www.cplusplus.com/doc/tutorial/operators/
// ↑ used to understand how operators work in code
// Source 2 - https://code4coding.com/c-example-to-print-elements-of-an-array/
// ↑ used because I forgot how to print arrays

#include <iostream>
#include <cstdio>
using namespace std;

int main ()
{
    //Declare Variables
    int a[8] = {0, 0, 0, 0, 1, 1, 1, 1};
    int b[8] = {0, 0, 1, 1, 0, 0, 1, 1};
    int c[8] = {0, 1, 0, 1, 0, 1, 0, 1};

    //List the Functions
    cout << "F1 = (A ∧ B) ∨ (A ∧ C)\t" << endl;
    cout << "F2 = (A ∧ C) ∧ (B ∧ ¬C)\t" << endl;
    cout << "F3 = (A ∨ B) ∧ ¬(B ∨ C)\t" << endl;
    cout << "F4 = (A ∨ (B ∧ C)) ∧ (¬A ∧ ¬B)\t" << endl;
```

```

cout << "F5 = ((B ∧ C) ∨ (C ∧ A)) ∧ (¬(A ∨ B) ∧ C)\n" << endl;

//Print the Header
cout << "A\tB\tC\tF1\tF2\tF3\tF4\tF5" << endl;;

//Output the answers
for (int i = 0; i < 8; i++)
{
    cout << a[i] << "\t" << b[i] << "\t" << c[i] << "\t";
    cout << ((a[i] && b[i]) || (a[i] && c[i])) << "\t";
    cout << ((a[i] && c[i]) && (b[i] && !c[i])) << "\t";
    cout << ((a[i] || b[i]) && ( !(b[i] || c[i])))) << "\t";
    cout << ((a[i] || (b[i] && c[i])) && ( !a[i] && (!b[i])))) << "\t";
    cout << (((b[i] && c[i]) || (c[i] && a[i])) && ( !(a[i] || b[i])) && c[i])) << "\t";
    cout << endl;
}
return 0;
}

```

Output:

```

[Running] cd "/Users/Dawlat/Desktop/CSC 1500/Lab/2,"
F1 = (A ∧ B) ∨ (A ∧ C)
F2 = (A ∧ C) ∧ (B ∧ ¬C)
F3 = (A ∨ B) ∧ ¬(B ∨ C)
F4 = (A ∨ (B ∧ C)) ∧ (¬A ∧ ¬B)
F5 = ((B ∧ C) ∨ (C ∧ A)) ∧ (¬(A ∨ B) ∧ C)

A    B    C    F1    F2    F3    F4    F5
0    0    0    0     0     0     0     0
0    0    1    0     0     0     0     0
0    1    0    0     0     0     0     0
0    1    1    0     0     0     0     0
1    0    0    0     0     1     0     0
1    0    1    1     0     0     0     0
1    1    0    1     0     0     0     0
1    1    1    1     0     0     0     0

[Done] exited with code=0 in 2.494 seconds

```