

## CSCI-14 Lab 5: Logic Practice, due 2/27/18

Hand-written exercises. You MAY NOT work alone on these. Do these alone first, then get with two or three other people to check each other's answers. Agree among yourselves what the answers are BEFORE you write test programs to verify the answers. Then write and turn in one paper solution with all team members' names on it. I do not want the test programs, just the team's paper solution. I suggest using truth tables to analyze many of these.

1. Write a simple expression that is equivalent to `!(A && B)` without using `!`
2. If you are `rich` then you are automatically `happy`. If you are not `rich` then you may still be `happy` but only if you are both `employed` and `healthy`. Complete the following assignment statement using `rich`, `employed` and `healthy` (all `bool` variables) along with one or more logical operators.

`happy =`

3. Which of the following are **not** equivalent to `!(cold && rainy)`? Hint: test your answers with a truth table.
  - a. `!(cold && rainy) || !rainy`
  - b. `!cold || !rainy`
  - c. `(!rainy || !cold) && (!cold || !rainy)`
  - d. `!!(!rainy || !cold)`
  - e. `!cold && !rainy`
4. Simplify `(big && red) || red`
5. Simplify `(big || red) && big`
6. Simplify `!(big || red) && big`
7. Assume that **A**, **B** and **C** are variables of type `bool`. Which of the following are **not** correct logical expressions in C++ (meaning they will not compile)?
  - a. `!(A) && (!B)`
  - b. `!(A || B)`
  - c. `A ! && B && C`
  - d. `!!A && !!B || !!C`
  - e. `!(!(!(A))) && !B`
8. Simplify the following if-else statement by using logical operators. You will only need one if-else. Assume that `A` and `B` are variables of type `bool`.

```
if (A)
    if (B)
        cout << "Fred";
    else
        cout << "Derf";
else
    cout << "Derf";
```

9. Simplify the following if-else statement by using logical operators. You will only need one if-else.

```
if (A)
    cout << "Man";
else if (B)
    cout << "Man";
else
    cout << "Van";
```

10. Specify the ranges of values of `n` (of type `int`) for which the following statement prints "Hello", "Goodbye" or nothing.

```
if (n < 10)
    if (n > 5)
        cout << "Hello";
else
    cout << "Goodbye";
```

11. Write the following if-else statement without using a "!" in the test condition:

```
if (!narcissist && !psychopath)
    cout << "friend material";
else
    cout << "run away!";
```

12. Simplify the following nested if-else statement into a single if-else. Do not use a "!" in your answer.

```
if (!famous)
    if (!rich)
        cout << "Unhappy";
    else
        cout << "Happy";
else
    cout << "Happy";
```

For each of the following pairs of expressions determine if they are "equivalent" or "not equivalent". Equivalent means they will produce the same columns of T/F values in truth tables. Test your answer by plugging in sample values. Careful: some of these are tricky.

13.        `a <= b`  
          `a < b && a == b`
14.        `a <= b || b <= c`  
          `a <= c`
15.        `(P || Q) && R`  
          `(P && R) || (Q && R)`
16.        `a == b`  
          `!(a > b || b > a)`

17.       $\neg(P \wedge Q)$   
          $\neg P \vee \neg Q$
18.       $P \wedge (Q \vee \neg P)$   
          $P \wedge Q$
19.       $\neg(\neg P \wedge Q)$   
          $\neg Q \vee P$
20.       $Q \vee P \vee \neg Q$   
          $P \vee \neg P$
21.       $\neg(\text{big} \vee \text{round})$   
          $\neg \text{big} \vee \neg \text{round}$
22.       $(\text{big} \wedge \text{round}) \vee (\text{big} \wedge \neg \text{round})$   
          $\text{big}$