CS 2102 - DMT1 - Spring 2020 — Luther Tychonievich
Practice exercise in class friday january 24, 2020

Practice 01

problem group 1 *English* $\mapsto$ *Math*

Rewrite each of the following English sentences as an expression over propositions. Include both a mapping from symbols to propositions and the final expression.

1. Jim Ryan will have to give up being the president of UVA if Teresa Sullivan returns to UVA

S: Sullivan returns to UVA
R: Ryan gives up presidency

$S \to R$

2. I prefer oranges to apples, although apples are less messy to eat

P: I prefer oranges to apples
M: apples are less messy than oranges

$P \wedge M$

3. If I forget my keys I can't get into the house unless my roommate is home.

K: I remember my keys
H: I can enter my house
R: My roommate is home

$(\neg K \wedge \neg R) \to \neg H$

4. If you can prove $P \neq NP$ (or $P = NP$, though I hope you don't), you'll become famous and I'll give you an A in this class

E: You can prove $P = NP$
N: You can prove $P \neq NP$
F: You'll be famous
A: I'll give you an A
H: I hope E

$((E \vee N) \to (F \wedge A)) \wedge \neg H$

5. Python programmers must be lazy because Python programs are so much shorter than the equivalent Java or C++ programs

L: Python programmers are lazy
J: Python programs are shorter than Java programs
C: Python programs are shorter than C++ programs

$(J \wedge C) \to L$

6. Python programs are so much shorter than the equivalent Java or C++ programs because Python programmers are lazy

L: Python programmers are lazy
J: Python programs are shorter than Java programs
C: Python programs are shorter than C++ programs

$L \to (J \wedge C)$

7. Excessive bail shall not be required, nor excessive fines imposed, nor cruel and unusual punishments inflicted

B: excessive bail required
F: excessive fines imposed
C: cruel punishments inflicted — *or* — P: cruel and unusual punishments inflected
U: unusual punishments inflicted

$\neg B \wedge \neg F \wedge \neg(C \wedge U)$ — *or* — $\neg B \wedge \neg F \wedge \neg P$

8. Because we know that no general-purpose sorting algorithm can be faster than $O(n \log n)$, if you hear about any faster algorithm you can know it must be "cheating" somehow

K: We know no general-purpose sorting algorithm can be faster than $O(n \log n)$
H: you hear of a faster algorithm
C: "faster" algorithm is cheating

$K \rightarrow (H \rightarrow C)$

PROBLEM GROUP 2 *If Statements*

Write an expression for when the following function returns the given return values. Use the parameters of the function as your propositions. Each function is shown in both Java and Python.

```
def f(a,b,c):
    if a or b:
        return "one"
    elif c != a:
        return "two"
    else:
        return "three"
```

```
public static String f(boolean a, boolean b, boolean c){
    if(a || b)
        return "one";
    else if(c != a)
        return "two";
    else
        return "three";
}
```

9. `f` returns `"one"` when $a \vee b$

10. `f` returns `"three"` when $\neg(a \vee b) \wedge \neg(c \oplus a)$ — or — $\overline{a} \wedge \overline{b} \wedge \overline{c}$

```
def g(a,b):
    if a:
        return "left"
    elif b:
        return "right"
    else:
        return "up"
```

```
public static String g(boolean a, boolean b){
    if(a)
        return "left";
    else if(b)
        return "right";
    else
        return "up";
}
```

11. g returns `"right"` when $b \wedge \neg a$

12. g returns `"up"` when $\neg b \wedge \neg a$

PROBLEM GROUP 3 *Truth Tables*

Fill in the following truth tables

13.

A	B	C	$(A \vee C) \leftrightarrow (B \wedge C)$		
0	0	0	0	1	0
0	0	1	1	0	0
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	1	1

14.

A	B	C	$\big((A \oplus B) \vee (A \oplus C)\big) \vee (B \oplus C)$				
0	0	0	0	0	0	**0**	0
0	0	1	0	1	1	**1**	1
0	1	0	1	1	0	**1**	1
0	1	1	1	1	1	**1**	0
1	0	0	1	1	1	**1**	0
1	0	1	1	1	0	**1**	1
1	1	0	0	1	1	**1**	1
1	1	1	0	0	0	**0**	0

15.

A	B	C	$(A \oplus C) \leftrightarrow (B \leftrightarrow C)$		
0	0	0	0	0	1
0	0	1	1	0	0
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	1	1	1
1	0	1	0	1	0
1	1	0	1	0	0
1	1	1	0	0	1