# CPEG 422/622 Spring 2020

# Homework 1

**Due February 24th at midnight (through Canvas)**
**Put your name in the comment part of the code you submitted!**

1. Analyze the 4-bit carry-ripple adder and the 4-bit carry look ahead (CLA) adder discussed in lecture 3 and fill in the following table about *gate count*. Show your analysis for partial credit.

| | 4-bit carry ripple adder | 4-bit CLA adder |
|---|---|---|
| XOR gates | 8 | 8 |
| AND gates | 8 | 11/14 |
| OR gates | 4 | 4 |
| Total gates | 20 | 23/26 |

Answer: each full adder has two 2-input XOR, two 2-input AND, one 2-input OR. The carry ripple adder is just the sum of four full adders.
The CLA adder has four half adders (each has one 2-input XOR and one 2-input AND), one CLA logic (has 6 AND gates and 3 OR gates for computing c1, c2, c3, plus one AND and one OR for computing c4 based on c3), and four 2-input XOR gates for computing sum.
If we fully expend the CLA logic for c4, we will need 10 AND gates and 4 OR gates for computing c1~c4.
40/40

2. Change the 4-bit carry-ripple adder VHDL code to implement a 4-bit carry look ahead (CLA) adder in the way described in lecture slides. Submit your VHDL code.

Grading policy:
Adder entity has A[3:0], B[3:0], Cin, Cout, and S[3:0], can use other names (e.g., X, Y) – 10
No syntax error – 10
CLA equations are correct – 20

$C_4 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0C_0$ // Or $C_4 = G_3 + P_3C_3$

$C_3 = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_0$

$C_2 = G_1 + P_1G_0 + P_1P_0C_0$

$C_1 = G_0 + P_0C_0$

Adder function is correct – 20