CSE3015 Fall 2020 Midterm Exam

1) (25 pts.) Suppose you have a **3-digit input X**, X={a b c} where each digit can be either 0 or 1. Create a circuit that detects whether the input is a **palindrome** or not.
(Palindrome: the input should be read same backward as forward)

a) (16 pts.) Start with a **truth table** and give the **equation**.

b) (9 pts.) Implement the logic function described above using: a **3-to-8 decoder** and an **OR gate**. Draw the wires from the outputs of the decoder to the inputs of the OR gate as necessery to do the job. You can assume you have as many inputs to the OR gate as you find necessary.

2) (25 pts.) A logic circuit has a 4-bit input represented as **X={a b c d}** and one output y, satisfying the following:

$$\mathbf{y=}\begin{cases}\mathbf{0, \text{ if } X<=4} \\ \mathbf{1, \text{ if } 5 <= X <=9} \\ \mathbf{X, \text{ if } X>=10}\end{cases}$$

a) (10 pts.) Find the **simplified** sum of products function for y.

b) (3 pts.) Draw its circuit.

c) (12 pts.) How would you **reduce** the number of **transistors** of the circuit in (b)? **Draw your new circuit. Indicate number of transistors used in your previous (b) and new circuit. What is the gate delay of your previous (b) and new circuit?**

3) (25 pts.) Given the incompletely specified function of four variables, where **d(A,B,C,D)** denotes **don't care** cases.

F(A,B,C,D)=∑(4,5,6,7,11,12,15) + d(A,B,C,D) = ∑(0,3,9,13).

a) (10 pts.) Find the **minimal sum of products** expression. **Do simplification.**

b) (15 pts.) Find the **minimal product of sums** expression. **Do simplification.**

4) (25 pts.) Assume we will read **two binary values x and y** (x and y can be either ‘0’ or ‘1’), where x is the value generated by a sensor and y is the value generated by an algorithm. Our objective is to **compare these two values x and y.**

We will **record the values x and y every 10 minutes** and we will use **the last 3 recorded values** (present, 10 minutes ago and 20 minutes ago) as follows:

Produce an output z, which will be **equal to either the x or y value recorded 20 minutes ago**. **If the current and previous recorded values of x and y are equal, then z will be equal to x, otherwise to y.**

Ex1:

If the stored x and y values are as in the following table, both the current and 10 min. ago values are the same for x and y, so z is equal to the value of x stored 20 min. ago, that is z is 1.

	Current	**10 min. ago**	**20 min. ago**
X	**0**	**1**	**1**
Y	**0**	**1**	0
	=	**=**	

Ex2:

If the stored x and y values are as in the second table, the current values are equal but 10 min. ago values are not, so z is equal to the value of y stored 20 min. ago, that is z is 0.

	Current	**10 min. ago**	**20 min. ago**
X	**0**	**0**	1
Y	**0**	**1**	**0**
	=	**≠**	

Do not forget that we have a stream of x and y values, so make sure that you have a design that updates values every 10 minutes.