CS226 Quiz 1 (Spring 2018)

Max marks: 15 **Time: 45 mins**

- *Be brief, complete and stick to what has been asked.*
- *Unless asked for explicitly, you may cite results/proofs covered in class without reproducing them.*
- *If you need to make any assumptions, state them clearly.*
- ***Please start writing your answer to each sub-question on a fresh page. DO NOT write answers to multiple sub-questions on the same page.***
- ***IIT Bombay prohibits the use of communication devices and internet enabled devices during examinations. You will be debarred from taking the examination if you are found accessing the internet during the examination.***
- ***Please do not engage in unfair or dishonest practices during the examination. Anybody found indulging in such practices will be referred to the D-ADAC.***

1. *[5 marks]* Draw an ROBDD for the function $(\overline{x_1} + x_2) \cdot (\overline{x_2} + x_3) \cdot (\overline{x_3} + x_4) \cdot (\overline{x_4} + x_1)$ using the variable order $x_4 < x_1 < x_2 < x_3$.

2. *[10 marks]* You are required to design a circuit that must be used to decide which of two users requesting access to a shared resource gains access to the resource. Specifically, assume that user 1 controls an input r_1, and user 2 controls an input r_2. User 1 sets r_1 to 1 whenever she wants to access the shared resource, and sets r_1 to 0 whenever she no longer needs access to the resource. The case for user2 is analogous. Your circuit must have two ouptuts, named g_1 and g_2. User 1 (respectively, user 2) gains access to the shared resource iff g_1 (respectively, g_2) is set to 1.

   Your circuit must satisfy the following properties:

   - g_1 and g_2 must never be set to 1 at the same time.
   - g_1 must not be set to 1 if r_1 is 0.
   - g_2 must not be set to 1 if r_2 is 0.
   - If r_1 and r_2 are both set to 1, then either g_1 or g_2 can be set to 1.
   - The circuit must grant access to the two users in a fair manner. Specifically, suppose both users simultaneously request access to the resource now, and suppose the circuit grants access to user 1. Then
     - the next time both users simultaneously request access (of course, this can happen only after one of the users has reset her request in between), the circuit must grant access to user 2;
     - if both users simultaneously request access again later (i.e. the third time), the circuit must then grant access to user 1.

   You may assume that gate delays are very small (negligible) compared to the delay between change of values of r_1 or to the delay between change of values of r_2. Thus, r_1 and r_2 by themselves change very slowly compared to the circuit delays. Assume also that either r_1 and r_2 change together (at the same time) or the time separation between a change of r_1 and a change of r_2 (or vice versa) is large compared to the circuit delays.

   Design a circuit (without latches/flip-flops) to implement the above functionality. Your circuit should take inputs r_1 and r_2 and produce outputs g_1 and g_2.

   Clearly explain your steps and justification. Your answer will be graded based on how you arrived at a description of the functions g_1 and g_2, and how you went about implementing them using K-maps, truth-tables or an interconnection of gates.