**Practice Problem Set 5**
Sharing & Binding, Datapath Controller/Generation

1. Given the following conflict graph in Fig. 1, how many ALUs and multipliers are needed? *(Assume vertices 1, 2, 3 are ALU operations, the remaining are multiple operations)*

<table>
<thead>
<tr>
<th>Number of ALUs required</th>
<th>Number of Multipliers required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Fig. 1: Conflict graph for Problem 1.](image)

2. Assuming the compatibility graph provided in Fig. 2, which of the following are valid cliques
   (a) \{1\}
   (b) \{1, 3, 8\}
   (c) \{3, 8, 7, 2\}
   (d) \{1, 2, 3, 6, 7, 8\}
   (e) None of the above

![Fig. 2: Compatibility graph for Problem 2.](image)

3. Given the provided scheduled sequencing graph, determine the minimum number of multipliers and registers that are needed along with the resulting bindings by using the following options,
   (a) Provide the corresponding *compatibility graph* and trace through the execution of the [*CLIQUE_PARTITION* algorithm].
   (b) Provide the corresponding *conflict graph* and trace through the execution of the [*VERTEX_COLOR* algorithm].
   (c) Provide the corresponding *interval graphs* (i.e. the graphical representation for the intervals as well as L) and trace through the execution of the [*LEFT_EDGE* algorithm].

![Time 1](image)
![Time 2](image)
![Time 3](image)
![Time 4](image)

4. Using the Left Edge algorithm, determine how many registers and are needed and the resulting register bindings for the provided scheduled sequencing graph. *Note/Hint: Be sure to clearly label the required edges of the sequencing graph.*

![Time 1](image)
![Time 2](image)
![Time 3](image)
![Time 4](image)

5. Complete the following steps to convert the code snippet provided to hardware implementation
   (a) Create a CDFG
   (b) Convert to a (unscheduled) sequencing graph
   (c) Schedule the sequencing graph assuming you have an upper latency bound of 4, multipliers and ALUs both require 1 cycle. *(Hint: don’t use force directed)*
   (d) Perform resource sharing/binding with the left edge algorithm
   (e) Perform register sharing/binding with the left edge algorithm
   (f) Create the corresponding datapath and controller (fsm)

```plaintext
inputs: t1, t2, t3, t4, t5, t6
outputs: a, b, c, d
a = t1 + t2 + t3
b = t2 * t4
c = a < b
d = (t5 + t5) >> t6
```