1. [5 points] Demonstrate what happens when we insert the keys 5, 28, 19, 15, 20, 33, 12, 17, 10 into a hash table with collision resolved by chaining. Let the table have 9 slots, and let the hash function be \( h(k) = k \mod 9 \).

2. [10 points] Give examples of graphs with the following requirements. You can describe them, or draw some illustration to help describe them.
   
   (1.a) Give an example of a graph with \( n \) nodes, and \( \Theta(n^2) \) number of edges.
   
   (1.b) Give an example of a DAG (directed acyclic graph) with \( n \) nodes, and \( \Theta(n^2) \) number of edges.

3. [5 points] Can the vertices of the graph in Figure 3 be topologically sorted?
   
   - If your answer is yes, give the graph vertices in a topologically sorted order.
   - If your answer is no, explain why not.

![Figure 3](image)

4. [15 points] Consider the following procedure \( \text{Proc1}(G) \) whose input is an undirected graph \( G \). Edges of \( G \) are represented by an adjacency LIST.

```plaintext
procedure \text{Proc1}(G) 
1 Q.Init(); /* Q is a priority queue implemented as a Min-heap */ 
2 foreach vertex \( v_i \in V(G) \) do 
3     foreach edge \((v_i, v_j)\) incident on \( v_i \) do 
4         foreach vertex \( v_p \in V(G) \) do 
5             \( s \leftarrow 0 ; \)
6             foreach edge \((v_p, v_q)\) incident on \( v_p \) do 
7                 if \( v_j = v_q \) then 
8                     \( s \leftarrow s + 1; \)
9                 end 
10             end 
11             Q.Insert(s); 
12         end 
13     end 
14 end
```

What is the maximum size of the priority queue \( Q \)? Analyze the running time of this algorithm in terms of the number of vertices \( n = |V| \) and the number of edges \( m = |E| \). (Justify your answer. Do not simply give a running time.)
procedure Proc2(G)  /* Q is a priority queue implemented as a Min-heap */
1 Q.Init();
2 foreach vertex v_i ∈ V(G) do
3     foreach vertex v_j ∈ V(G) do
4         if (A[i,j] == 1) then
5             x ← i * n + j;
6             Q.Insert(x);
7         end
8     end
9 end
10 while (Q.IsNotEmpty()) do
11     x ← Q.DeleteMin();
12     Print x;
13 end

5. [15 points] Consider the following procedure (Proc2(G)) whose input is an undirected graph G. Edges of G are represented by the adjacency MATRIX A of G.
   Analyze the running time of this algorithm in terms of the number of vertices n = |V| and the number of edges m = |E|. (Justify your answer. Do no simply give a running time.)

6. [15 points] There are two types of professional wrestlers: “babyfaces” (“good guys”) and “heels” (“bad guys”). Between any pair of professional wrestlers, there may or may not be a rivalry. Suppose we have n professional wrestles \(V = \{v_1, \ldots, v_n\}\) and we have a list of m pairs of wrestlers for which there are rivalries. Give an \(O(n + m)\) time algorithm that determines whether it is possible to designate some of the wrestlers as babyfaces and the remainder as heels such that each rivalry is between a babyface and a heel. If it is possible to perform such a designation, your algorithm should produce it.