Given a set of \( n \) specimens and \( m \) judgments, we need to determine if the set of judgments are consistent. To be able to label each specimen either \( A \) or \( R \), we construct an undirected graph \( G = (V, E) \) as follows: Each specimen is a vertex. There is an edge between \( v_i \) and \( v_j \) if there is a judgment involving the corresponding specimens.

Once the graph is constructed, arbitrarily designate some vertex as the starting node \( s \). Note that the graph \( G \) need not be connected in which case we will need starting nodes for each component. For each component \( G_i \) (with starting node \( s_i \)) of \( G \) label the specimen associated with \( s_i \) \( A \). Now perform Breadth-First Search on \( G_i \) starting at \( s_i \). For each node \( v_k \) that is visited from \( v_j \), consider the judgment made on the specimens corresponding to \((v_j, v_k)\). If the judgment was "same," label \( v_k \) the same as \( v_j \) and if the judgment was "different," label \( v_k \) the opposite of \( v_j \). Note that there may be some specimens that are not associated with any judgments. These specimens may be labeled arbitrarily, but we shall label them \( A \). Once the labeling is complete we may go through the list of judgments to check for consistency. More precisely (Refer to pg. 39 of the text)

For each component \( C \) of \( G \)
- designate a starting node \( s \) and label it \( A \)
- Mark \( s \) as "Visited."
- Initialize \( R=s \).
- Define layer \( L(0)=s \).
- For \( i=0,1,2,... \)
  - For each node \( u \) in \( L(i) \)
    - Consider each edge \((u,v)\) incident to \( v \)
    - If \( v \) is not marked "Visited" (then \( v \) is also not labeled)
      - Mark \( v \) "Visited"
      - If the judgment \((u,v)\) was "same" then
        - label \( v \) the same as \( u \)
      - else (the judgment was "different")
        - label \( v \) the opposite of \( u \)
    - Endif
  - Add \( v \) to the set \( R \) and to layer \( L(i+1) \)
  - Endif
- Endfor
Endfor

For each edge \((u,v)\) (for each judgment \((u,v)\))
- If the judgment was "same"
  - If \( u \) and \( v \) have different labels
    - there is an inconsistency
  - Endif
Else (the judgment was "different")

\(^1\text{cx}926.863.309\)
If \( u \) and \( v \) have the same labels
there is an inconsistency
Endif
Endif
Endfor

First note that the running time of this algorithm is \( O(m + n) \): Constructing \( G \) takes \( O(m + n) \) since it has \( n \) vertices and \( m \) edges. Performing \( BFS \) on \( G \) takes \( O(m + n) \) and going through the list of judgments to check consistency takes \( O(m) \). Thus the running time is \( O(m + n) \).

It is easily shown that if the labeling produced by the \( BFS \) is inconsistent, then the set of judgments is inconsistent. Note that this \( BFS \) labeling uses a subset of the judgments (the edges of the resulting \( BFS \) tree). Further the \( BFS \) labeling is the only possible labeling with the exception of inverting the labeling in each component of \( G \), i.e. switching \( A \) and \( B \). Thus if an inconsistency is found in this labeling then surely the entire set of \( m \) judgments cannot be consistent. On the other hand if the labeling is consistent with respect to the \( m \) judgments, we are done.