Question 1: Hypergraph Acyclicity

Let $H$ be a hypergraph. Prove that the following are all equivalent:

1. $H$ has a join tree.
2. There exists a sequence of ear removals on $H$ wherein all hyperedges of are eliminated.
3. Every sequence of ear removals (including the empty one) can be completed into one that eliminates all hyperedges.

Explain how one can detect in polynomial time whether a given hypergraph is acyclic.

Question 2: Yannakakis’s Algorithm

In this question we will prove the correctness and efficiency of Yannakakis’s join algorithm. Let $\alpha = \pi_A(R_1 \bowtie \cdots \bowtie R_k)$ be an acyclic CQ expression, and $T$ be a rooted (directed) join graph for $H_\alpha$. For $i = 1, \ldots, k$, let $r_i$ be a relation over $R_i$, and let $r_i'$ be the relation that remains from $r_i$ following the inside-out phase.

1. Prove the following:
   
   (a) $r_1 \bowtie \cdots \bowtie r_k = r_1' \bowtie \cdots \bowtie r_k'$.
   
   (b) Every fact (tuple) of every $r_i'$ participates in (i.e., can be joined with) at least one tuple of $r_1 \bowtie \cdots \bowtie r_k$.

   Hint 1: Use induction based on $T$. Hint 2: The inductive step does not have to be from children to parents (as a typical tree based induction).

2. Prove that for all nodes $v$ of $T$ it holds that $\text{result}(v) = \pi_{O_i, P_i}(r_1 \bowtie \cdots \bowtie r_k)$. Hint 1: Use induction based on $T$. Hint 2: Now children-to-parent should work.

3. Conclude that Yannakakis’s algorithm correctly computes $\pi_A(r_1 \bowtie \cdots \bowtie r_k)$.

4. Prove that the algorithm terminates in polynomial total time.

5. Where in the above proofs did you use the fact that $T$ is a join tree (and not just a tree of relations)?

Good luck!