



**CALIFORNIA STATE SCIENCE FAIR  
2016 PROJECT SUMMARY**

<b>Name(s)</b> <b>Lev Kruglyak</b>	<b>Project Number</b> <b>J1417</b>
<b>Project Title</b> <b>Inductive Reasoning Algorithm</b>	
<b>Abstract</b> <b>Objectives/Goals</b> The objective of this study is to understand whether or not the algorithm I created could be used to solve geometry problems. <b>Methods/Materials</b> Computer with Eclipse IDE and JRE 1.8. The algorithm using a recursive function to build an inverse inductively defined tree and then traverses it to find the optimal solution. Tested the algorithm on varying types of triangle congruence problems. <b>Results</b> Algorithm found the optimal solution in a short amount of time for standard triangle congruence problems. Algorithm was less efficient in solving problems with lots of misleading turns in the logic. Algorithm was incredibly good at separating the optimal solution from a problem with lots of extra information. <b>Conclusions/Discussion</b> The Inductive Reasoning Algorithm however, will never work on problems that require the use of algebraic properties, such as transitive property, substitution property, etc. The reason for this is the way the algorithm solves problems. The algorithm proves the problem by building a tree from the prove statement to the given using rules that describe a transformation from one set of object relationships to another, and the input can be generated from the output. Properties also describe a transformation from input to output, but the input cannot be generated from the output. Because of this, the Geometry Bot Algorithm cannot use properties to solve a problem. An example of this is the transitive property. The transitive property states that if $a = b$ , and $b = c$ , then $a = c$ . Since the algorithm solves everything backwards, the algorithm would be asked to prove $a = c$ . The algorithm has no way of knowing that $b$ is the common term, so it would try to prove that $a$ and $c$ are congruent to every other value in the problem, leading to an infinitely large problem.	
<b>Summary Statement</b> I devised an algorithm for solving problems using Inductive Reasoning, that can quickly find an optimal solution using a database of rules.	
<b>Help Received</b>	