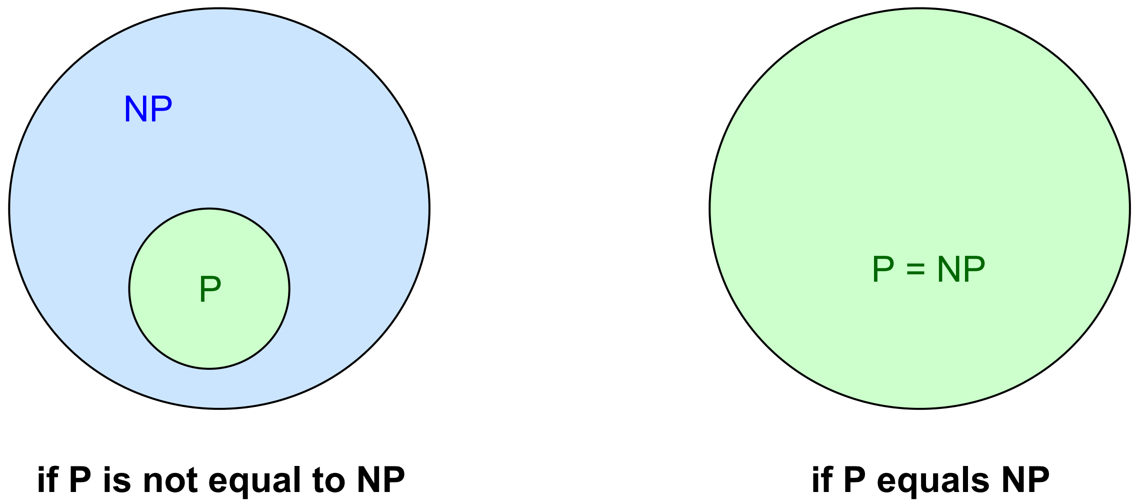
Get Rich with Complexity

In this lesson, we discuss the famous is P=NP dilemma.

P equals NP or P doesn't equal NP?

Since we can solve problems in P in polynomial time, we can also verify them in polynomial time. Thus we can say ***every problem in P is also in NP***. Note we haven't made the converse claim whether problems in **NP** are in **P** or not. Or said another way, is **P** a proper or improper subset of **NP**? We'll come back to that later.

One of the most important open questions in theoretical computer science is whether **P**=**NP**. It's so important and famous that Clay Institute includes it as one of the millennium problems, offering $1 million dollars in prize money for a solution! If **P** equals **NP** then the two classes would collapse into one. A pictorial representation is shown below



P vs NP

It's very important to understand that just because we can't prove **P**=**NP**, it doesn't imply that **P≠NP**. We require a proof to conclusively declare one or the other. Most theorists, however, believe that **P≠NP**. If it were proved **P**=**NP**, then it would mean that all problems that we can't solve today in polynomial time will have efficient solutions.