

## Notes Week 9

- Exam Week 10 - B-M 6:45pm, N-Y 6:00pm

## Assignment - Minimum Dominating Set of a Graph

- Big ole Assignment
  - Exploring  $2^n$  runtime complexity
  - Writing a paper that:
    1. The problem
    2. The data
    3. The algorithm

## Minimum Domating Set of a Graph

- The nodes of a dominating set may not be unique
- The size of the set of nodes *will* be unique
- **Dominating Set of a Graph**: a subset of the vertices such that all nodes in the graph are either in the dominating set or have a neighbor that is in the dominating set.
- The best *known* solution to this problem is to test every possible combination of verticecs, aka **Brute Force**
- Use an array that is one larger than the number of nodes
- Using binaries numbers, can represent whether a node appears in the set using 0s or 1s.
- For each number 0 to  $2^n$ , pass the number to a function to determine if that number represents a dominating set
- **Approximation Algorithm**: an algorithm we design to run quickly trying to get an answer that's close to the correct answer

## Complexity Classes

- **P**: the set of all Problems that can be solved in polynomial times - constant exponents
- **NP**: the set of all problems whose solutions can be *verified* in polynomial time
- $P \in NP$
- **NP-Complete**: The hardest problems in **NP**.