Problem Definition:
Constraint Satisfaction Problem:

A constraint satisfaction problem**(CSP)** consists of three components, **X**, **D**, and **C**:
X is a set of variables, $\{X_1, \ldots, X_n\}$.
D is a set of domains, $\{D_1, \ldots, D_n\}$, one for each variable.
C is a set of constraints that specify allowable combinations of values.

Each domain D_i consists of a set of allowable values, $\{v_1, \ldots, v_k\}$ for variable X_i. Each constraint C_i consists of a relation between two variables. This type of constraint is called binary constraint

An assignment that does not violate any constraints is called a consistent or legal assignment. A **complete assignment** is one in which every variable is assigned, and a solution to a **CSP** is a **consistent**, **complete assignment**.

A variable in a CSP is **arc-consistent** if every value in its domain satisfies the variable's binary constraints.

X_i is arc-consistent with respect to another variable X_j if for every value in the current domain D_i there is some value in the domain D_j that satisfies the binary constraint on the arc (X, X_j). A network is **arc-consistent** if every variable is arc-consistent with every other variable.

In this problem we need to implement the most popular algorithm for arc consistency named AC-1, AC-2, AC-3, AC4.

Experimental Setting(Plan):

1. A set of constraints which will contain some relations between two variables of the consistency graph. The relations could be,

$$x \le y$$
$$x \ge y$$
$$y = x^2$$
$$\gcd(x,y) = 1$$
$$y \% x = 0$$

2. We will generate random graph using some python library named NetworkX for different number of nodes (e.g. 10,20,30,40,50)

3. For each node, we will consider it as a variable and randomly choose it's domain from a set of integers in a range(e.g.1~100) so that we can test the mathematical relational constraints. For each pair of variable we will choose one or more binary constraint from our constraint list(1).