**Easy Tasks (60-74 points)**

**Task 1: Reflexivity Checker** Objective: Write a program that checks if a relation is reflexive on a set.

* Input:
  + Set: {1, 2, 3}
  + Relation: {(1, 1), (2, 2), (3, 3)}
* Output:
  + Is Reflexive: True

**Task 2: Symmetry Identifier** Objective: Implement a program that identifies if a relation on a set is symmetric.

* Input:
  + Relation: {(1, 2), (2, 1), (3, 3)}
* Output:
  + Is Symmetric: True

**Task 3: ] Verifier** Objective: Create a program to verify the transitivity of a relation on a set.

* Input:
  + Relation: {(1, 2), (2, 3), (1, 3)}
* Output:
  + Is Transitive: True

**Medium Tasks (75-89 points)**

**Task 4: Equivalence Relation Checker** Objective: Create a program that checks if a relation on a set is an equivalence relation (reflexive, symmetric, and transitive).

* Input:
  + Set: {1, 2, 3}
  + Relation: {(1, 1), (2, 2), (3, 3), (1, 2), (2, 1), (2, 3), (3, 2)}
* Output:
  + Is Equivalence Relation: False

**Hard Tasks (90-100 points)**

**Task 5: Inverse Relation Generator** Objective: Implement a program that generates the inverse of a given relation.

* Input:
  + Relation: {(1, 2), (3, 4), (5, 6)}
* Output:
  + Inverse Relation: {(2, 1), (4, 3), (6, 5)}