**Homework 2**

**Name:** Chhay Lay Heng

**Net ID:** cxh220032

1. Determine relations on the set of all people are equivalence relations.
   1. {(a, b) | a and b are the same age} 🡪 equivalence relation

Reflexive: Yes

Symmetric: Yes

Transitive: Yes

* 1. {(a, b) | a and b have the same parents} 🡪 Equivalance

Reflexive: Yes

* 1. {(a, b) | a and b share a common parent} 🡪 not equivalence relation

Reflexive: Yes

Symmetric: Yes

Transitive: No

* 1. {(a, b) | a and b have met} 🡪 not equivalence relation

Reflexive: Yes

Symmetric: Yes

Transitive: No

* 1. {(a, b) | a and b speak a common language} 🡪 equivalence relation

Reflexive: Yes

Symmetric: Yes

Transitive: Yes

1. Find equivalence classes of following relations if they exist.
   1. {(0, 0), (1, 1), (2, 2), (3, 3)}

[0] = {0}

[1] = {1}

[2] = {2}

[3] = {3}

* 1. {(0, 0), (0, 2), (2, 0), (2, 2), (2, 3), (3, 2), (3, 3)}

[0] = {0, 2}  
[2] = {0, 2, 3}

[3] = {2, 3}

* 1. {(0, 0), (1, 1), (1, 2), (2, 1), (2, 2), (3, 3)}

[0] = {0}

[1] = {1, 2}

[2] = {1, 2}

[3] = {3}

* 1. {(0, 0), (1, 1), (1, 3), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)}

[0] = {0}

[1] = {1, 3}

[2] = {2, 3}

[3] = {1, 2, 3}

* 1. {(0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (1, 2), (2, 0), (2, 2), (3, 3)}

[0] = {0, 1, 2}

[1] = {0, 1, 2}

[2] = {0, 2}

[3] = {3}

1. Find reflexive closure graph, symmetric closure graph and transitive closure graph for following graphs

Chart

Description automatically generated with medium confidence

* Reflexive closure:
* Symmetric closure: a,c) (c,b)}
* Transitive closure:

(a,b) (b,c) 🡪 (a,c)

(c,a) (a,b) 🡪 (c,b)

(b,c) (c,a) 🡪 (b,a)

Diagram

Description automatically generated

* Reflexive closure:
* Symmetric closure:
* Transitive closure:

(a,c) (c,d) 🡪 (a,d)

(c,d) (d,b) 🡪 (c,b)

(b,a) (a,c) 🡪 (b,c)

1. Find the matrices

R1 = {(1,2) (2,1) (2,2) (2,3) (3,1)}

R2 = {(1,2) (2,2) (2,3) (3,1) (3,2) (3,3)}

* 1. {(1,2) (2,1) (2,2) (2,3) (3,1) (3,2) (3,3)}
  2. {(1,2) (2,2) (2,3) (3,1)}
  3. {(1,2) (1,3) (2,1) (2,2) (2,3) (3,2)}
  4. {(1,1) (1,2) (1,3) (2,1) (2,2) (2,3) (3,2)}
  5. {(2,1) (3,2) (3,3)}

1. Find