Question 1:

1. Using Warshall's algorithm, compute the reflexive-transitive closure of the relation below. Show the matrix after the reflexive closure and then after each pass of the outermost `for` loop that computes the transitive closure.

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Solution:

Reflexive Closure:

Reflexive Closure The reflexive closure of a matrix is obtained by **adding (a, a) to Matrix for each a $\in$ A where A is the relation which builds up the matrix**. where in this case A contains the column and row numbers. Thus, (1,1) denotes the intersecting element at 1st column and 1st row. (2,2) denotes element at 2nd column and 2nd row and so on.

- **Reflexive Closure Matrix:**

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

- **Transitive Closure Matrix:**

First Iteration:

In 1st iteration, 1st column and 1st row are selected and the positions of elements with 1 are placed in a set.

C = { 1 }

R = { 1 , 2 }

- **C x R = { (1,1) (1,2) }**

  - $$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Second Iteration:

In 2nd iteration, 2nd column and 2nd row are selected from the matrix obtained after 1st iteration and the positions of elements with 1 are placed in a set.

C = { 1 , 2}

R = { 2 , 4 }

- **C x R = { (1,2) (1,4) (2,2) (2,4) }**

  - $$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Third Iteration:

In 3rd iteration, 3rd column and 3rd row are selected from the matrix obtained after 2nd iteration and the positions of elements with 1 are placed in a set.

C = { 3 , 5}

R = { 3 , 4 }

- **C x R = { (3,3) (3,4) (5,3) (5,4) }**

  - $$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Fourth Iteration:

In 4th iteration, 4th column and 4th row are selected from the matrix obtained after 3rd iteration and the positions of elements with 1 are placed in a set.

C = { 1 , 2 , 3 , 4 , 5}

R = { 4 , 5 }

- **C x R = { (1,4) (1,5) (2,4) (2,5) (3,4) (3,5) (4,4) (4,5) (5,4) (5,5) }**

  - $$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Fifth Iteration:

In 5th iteration, 5th column and 5th row are selected from the matrix obtained after 4th iteration and the positions of elements with 1 are placed in a set.

C = { 1 , 2 , 3 , 4 , 5}

R = { 3 , 4 , 5 }

- **C x R = { (1,3) (1,4) (1,5) (2,3) (2,4) (2,5) (3,3) (3,4) (3,5) (4,3) (4,4) (4,5) (5,3) (5,4) (5,5) }**

  - $$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Above is the matrix obtained after 5th Iteration having Transitive Closure.

Question 2:

2. Using the matrix in the previous problem show the final result of executing Floyd's algorithm on that matrix to produce a matrix containing path lengths.

Solution:

Floyd's algorithm is a method for finding the shortest path between every pair of vertices in a weighted graph. In this case, we can use it to find the length of the shortest path between every pair of elements in the relation represented by the matrix.

The algorithm works by repeatedly updating the matrix with the minimum path lengths between pairs of vertices using a triple nested loop:

```
for k from 1 to |A|:
  for i from 1 to |A|:
    for j from 1 to |A|:
      if dist[i][j] > dist[i][k] + dist[k][j], set dist[i][j] to dist[i][k] + dist[k][j]
```

Here, `dist` is a matrix where `dist[i][j]` represents the shortest path length between vertices `i` and `j`. Initially, `dist` is the matrix representing the given relation, with `dist[i][j] = 1` if `(i, j)` belongs to the relation and `dist[i][j] = 0` otherwise.

Original Matrix :

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Converting the matrix to distance matrix form, where each adjacent node will have a distance of 1 and all other nodes ∞ .

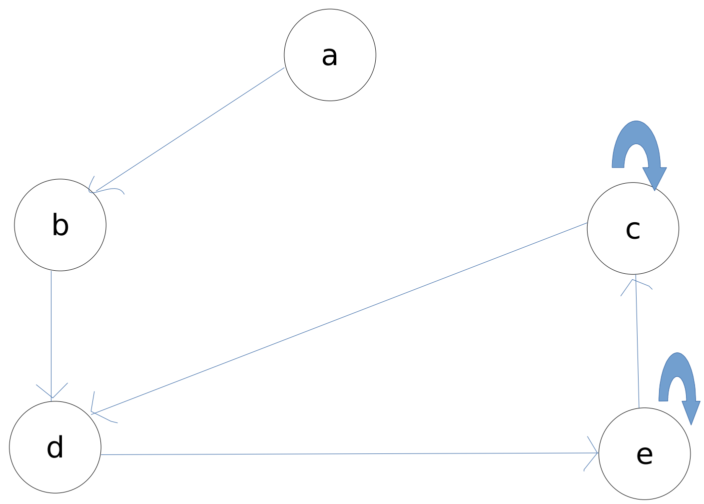
	1	**2**	**3**	**4**	**5**
1	0	1	∞	∞	∞
2	∞	0	∞	1	∞
3	∞	∞	0	1	∞
4	∞	∞	∞	0	1
5	∞	∞	1	∞	0

After the Five iteration of the algorithm, the matrix `dist` will be updated to final as:

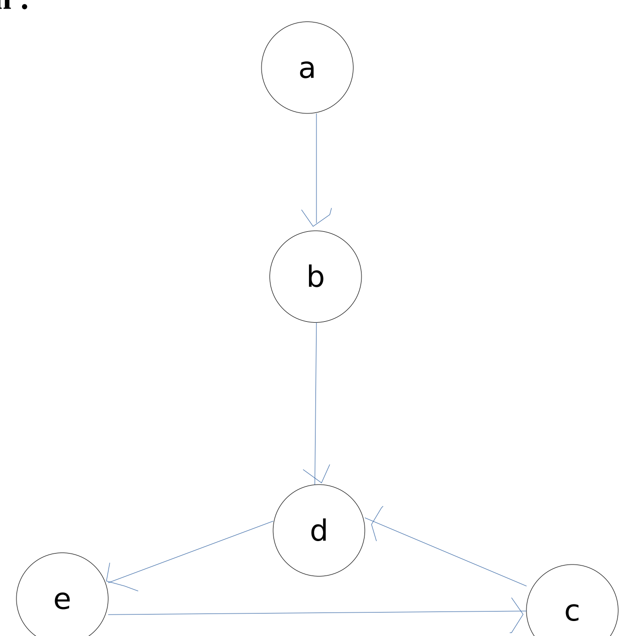
	1	**2**	**3**	**4**	**5**
1	0	1	4	2	3
2	∞	0	3	1	2
3	∞	∞	0	1	2
4	∞	∞	2	0	1
5	∞	∞	1	2	0

Question 3:

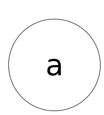
3. Show the graph that corresponds to the matrix in the first problem assuming the rows and columns correspond to the vertices a, b, c, d and e. Show its condensation graph, renaming its vertices. Determine any topological order of that graph and create an adjacency matrix with the vertices ordered in that topological order. Finally compute the reflexive-transitive closure of that matrix. What characteristic of that matrix indicates that it defines a total order?

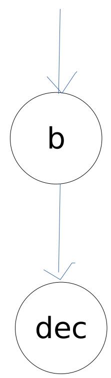


Condensation Graph :



which transforms into:





Topological order:

a → b → dec

Adjacency matrix:

	a	**b**	**dec**
a	0	1	0
b	0	0	1
dec	0	0	0

Reflexive – Closure :

	a	**b**	**dec**
a	1	1	0
b	0	1	1
dec	0	0	1

Transitive – Closure :

1^{st} Iteration:

In 1^{st} iteration, 1^{st} column and 1^{st} row are selected and the positions of elements with 1 are placed in a set.

C = { 1 }

R = { 1 , 2 }

- **C x R = {(1 ,1) (1,2)}**

	a	b	dec
a	1	1	1
b	0	1	1
dec	0	0	1

2^{nd} Iteration:

In 2^{nd} iteration, 2^{nd} column and 2^{nd} row are selected and the positions of elements with 1 are placed in a set.

C = { 2 ,3 }

R = { 2 , 3 }

- **C x R = { (2,2) (2,3) (3,2) (3,3)}**

	a	b	dec
a	1	1	1
b	0	1	1
dec	0	1	1

3^{rd} Iteration:

In 3^{rd} iteration, 3^{rd} column and 3^{rd} row are selected from the matrix obtained after 2^{nd} iteration and the positions of elements with 1 are placed in a set.

C = { 1,2,3}

R = { 2,3}

- **C x R = { (1,2) (1,3) (2,2) (2,3) (3,2) (3,3) }**

	a	b	dec
a	1	1	1
b	0	1	1
dec	0	1	1

Above is the matrix obtained after 3^{rd} Iteration having Reflexive-Transitive Closure.

- *It is a linear relation, and therefore a total order.*

Question 4:

4. Using Floyd’s algorithm, compute the distance matrix for the weight directed graph defined by the following matrix:

$$\begin{bmatrix} 0 & \infty & 4 & -2 \\ \infty & 2 & 3 & 6 \\ -3 & 2 & 0 & \infty \\ 4 & \infty & 5 & 0 \end{bmatrix}$$

Show the intermediate matrices after each iteration of the outermost loop.

Solution:

Matrix =

	a	**b**	**c**	**d**
a	0	∞	4	-2
b	∞	2	3	6
c	-3	2	0	∞
d	4	∞	5	0

After the first iteration of the algorithm, the matrix will be updated as follows:

	a	**b**	**c**	**d**
a	0	∞	4	-2
b	∞	2	3	6
c	-3	2	0	-5
d	4	∞	5	0

After the 2nd iteration of the algorithm, the matrix will be updated as follows:

	a	**b**	**c**	**d**
a	0	∞	4	-2
b	∞	2	3	6
c	-3	2	0	-5
d	4	∞	5	0

After the 3rd iteration of the algorithm, the matrix will be updated as follows:

	a	**b**	**c**	**d**
a	0	6	4	-2
b	0	2	3	-2
c	-3	2	0	-5
d	-2	7	5	0

After the 3rd iteration of the algorithm, the matrix will be updated as follows:

	a	**b**	**c**	**d**
a	0	5	3	-2
b	0	2	3	-2
c	-3	2	0	-5
d	-2	7	5	0

Above is the distance matrix obtained using Floyd's algorithm.