TRANSITIVE RELATIONS

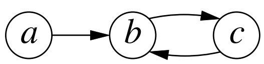
Transitivity.

- For all $x \neq y$ and $y \neq z$, if (x, y) and (y, z) are in R, then (x, z) is in R.

  Put another way, if one can go from x to z in two steps (via some y), then one must be able to go from x to z in one step.

Example.

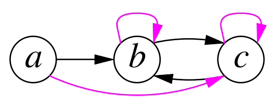
- The relation below violates the condition for transitivity in 3 ways as indicated:



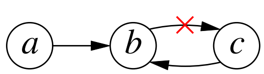
(a, c) should be in R because (a, b) and (b, c) are in R
(b, b) should be in R because (b, c) and (c, b) are in R
(c, c) should be in R because (c, b) and (b, c) are in R

Two Ways of Making Above R Transitive.

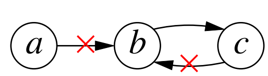
- Enlarge R by adding only the missing pairs or links $\{(a, c), (b, b), (c, c)\}$ as shown below.



- Reduce R to eliminate the cause(s) of each missing link, i.e., eliminate at least one link from each pair of links in "because ⋯" and avoid unnecessary link-elimination. Shown below are two such minimal reductions.



Remove (b, c)



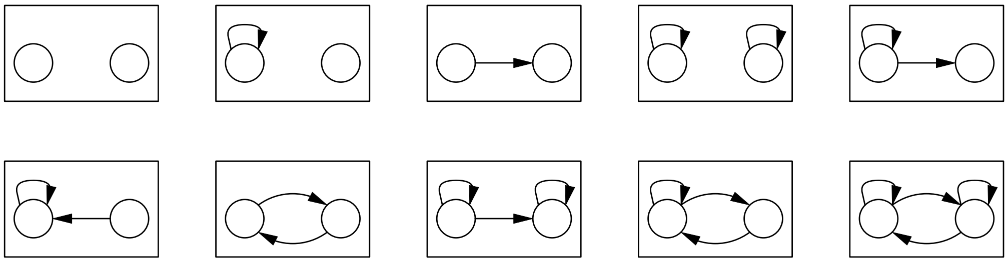
Remove (a, b) and (c, b)

Transitive Closure.

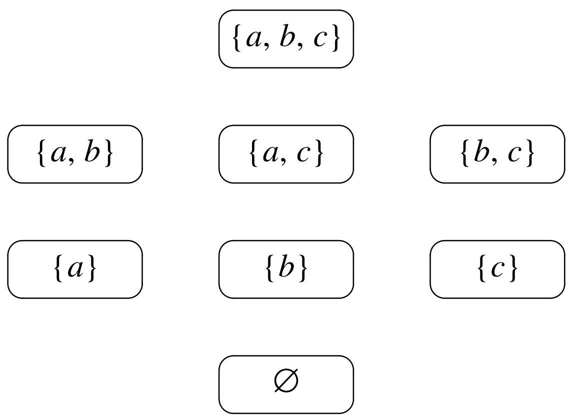
- If R is not transitive, then its transitive closure is obtained by adding the minimum number of links to R to make the enlarged relation, denoted by R^+, transitive.
- If R is already transitive, then $R^+ = R$.
- Because R^+ is always unique, which may not be the case when we reduce R to make it transitive, the standard way of making R transitive is to form R^+.

Practice Questions.

1. Shown below are the structures of all relations on $|X| = 2$. Mark those which correspond to a transitive relation; for the others, show the digraph of their transitive closure.



2. Let $S = \{a, b, c\}$ and $X = 2^S$, i.e., the set of all subsets of S. Consider the proper-subset relation "$\subset$" on X, i.e., for subsets $S_1, S_2 \subseteq S$ we have (S_1, S_2) is in $\subset$-relation if $S_1 \subset S_2$, i.e., $S_1 \subseteq S_2$ but $S_1 \neq S_2$. Draw the links below between the items of X for $\subset$-relation.



   Is $\subset$-relation transitive? How will the digraph change if we consider the subset-relation "$\subseteq$" and is $\subseteq$-relation transitive?

3. Show the digraph of a largest (in terms of #(links)) transitive relation on a set X of size 4. Do the same for a smallest transitive relation.

4. Argue that if R is transitive and we have a path $\langle x_1, x_2, \cdots, x_{k+1}\rangle$ of length $k \geq 2$ in the digraph of R, then $(x_1, x_{k+1}) \in R$.

   Likewise, argue that if R is transitive and we have a cycle $\langle x_1, x_2, \cdots, x_k, x_1\rangle$ of length $k \geq 2$ in the digraph of R, then $(x_1, x_1) \in R$.

5. Argue that if R is a symmetric relation, then its transitive closure R^+ is also a symmetric relation.

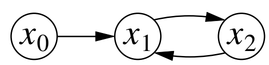
TESTING TRANSITIVITY PROPERTY OF A RELATION-MATRIX

Code for $n \times n$ matrix R.

```
for (i = 0; i < n; i++)
    for (j = 0; j < n; j++) //all (i, j) pairs
        if (0 == R[i][j]) //possible violation for link (i, j)
           for (k = 0; k < n; k++)
               if (2 == R[i][k] + R[k][j])
                  return(false);
return(true);
```

Performance.

- Consider the relation shown below.



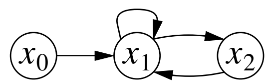
We write $R_{i,k}$ for $R[i][k]$, etc. The symbol "−" indicates the test is not done.

i	j	$R_{i,j}$	$2 == R_{i,0} + R_{0,j}$	$2 == R_{i,1} + R_{1,j}$	$2 == R_{i,2} + R_{2,j}$	Return-value
0	0	0	$2 == 0 + 0$ (F)	$2 == 1 + 0$ (F)	$2 == 0 + 0$ (F)	
0	1	1	−	−	−	
0	2	0	$2 == 0 + 0$ (F)	$2 == 1 + 1$ (T)	−	false

- If R is transitive, then for each $(x_i, x_j) \notin R$, all n iterations of k-loop will be done. Thus, the comparison "`2 == R[i][k] + R[k][j]`" will be done $n(n^2 - |R|)$ times.

Practice Questions.

1. Show the table of all comparisons done (and their T/F evaluations) for the relation shown below.



2. Following is a variation of the code for testing transitivity property of a symmetric relation-matrix R, where we only consider the pairs $i \leq j$, which makes this code more efficient than the original code.

```
for (i = 0; i < n; i++)
    for (j = i; j < n; j++)
        if (0 == R[i][j])
           for (k = 0; k < n; k++)
               if (2 == R[i][k] + R[k][j]) return(false);
return(true);
```

   Show the digraph of a suitable symmetric R for $n = 2$ to illustrate the need for initializing $j = i$ instead of $j = i+1$ in the j-loop.

3. Shown below is a code that creates the transitive closure of a relation-matrix R.

```
for (k = 0; k < n; k++)
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
            if (2 == R[i][k] + R[k][j]) R[i][j] = 1;
```

   (a) There is one source of inefficiency here in that for some (i, j)-pair the assignment `R[i][[j] = 1` may be done more than once. Modify the code to avoid this and explain whether this makes code necessarily more efficient or not.

   (b) Give the digraph of a relation R for which the following code does not work to create the transitive closure.

```
for (i = 0; i < n; i++)
    for (j = 0; j < n; j++)
        for (k = 0; k < n; k++)
            if (2 == R[i][k] + R[k][j]) R[i][j] = 1;
```