Foundations of Computer Science Comp109

University of Liverpool

Boris Konev

konev@liverpool.ac.uk

Olga Anosova

O.Anosova@liverpool.ac.uk

Recap: Cartesian product and binary relations.

- The *Cartesian product* $A \times B = \{(a, b) \mid a \in A \text{ and } b \in B\}$.
- A *binary relation* R between two sets A and B is $R \subseteq A \times B$, which can be represented by a *digraph*.
- Function $=$ *functional relation* defined on *all* elements.
- *Inverse relation* $R^{-1} = \{(b, a) \mid (a, b) \in R\}$.
- *Composition* $S \circ \underline{R} = \{(a, c) \mid \text{ exists } b \in B \text{ such that } aRb \text{ and } bSc\}$, **beware of the order!**

$f \circ (g(x))$

Computer friendly representation of binary relations: matrices

- Let $A = \{a_1, \ldots, a_n\}$, $B = \{b_1, \ldots, b_m\}$ and $R \subseteq A \times B$.
- We represent R by an array M of n rows and m columns. Such an array is called a *n by m matrix*.
- The entry in row i and column j of this matrix is given by $M(i,j)$ where

$$M(i,j) = \begin{cases} 1 & \text{if} \quad (a_i, b_j) \in R \\ 0 & \text{if} \quad (a_i, b_j) \notin R \end{cases}$$

Example 1

Let $A = \{1, 3, 5, 7\}$, $B = \{2, 4, 6\}$, and

$$U = \{(x, y) \in A \times B \mid x + y = 9\}$$

Assume an enumeration $a_1 = 1$, $a_2 = 3$, $a_3 = 5$, $a_4 = 7$ and $b_1 = 2$, $b_2 = 4$, $b_3 = 6$. Then M represents U, where

$$M = \begin{array}{c} \\ 1=a_1 \\ 3 \\ 5 \\ 7 \end{array} \begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \end{array}$$

$1 + 2 \overset{?}{=} 9$

$3 + 6 = 9$

Example 2

Let $A = \{a, b, c, d\}$ and suppose that $R \subseteq A \times A$ has the following matrix representation:

$$M = \begin{array}{cc} & \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

List the ordered pairs belonging to R.

$\{(a,b), (a,c), \ldots$

Example

The binary relation R on $A = \{1, 2, 3, 4\}$ has the following digraph representation.

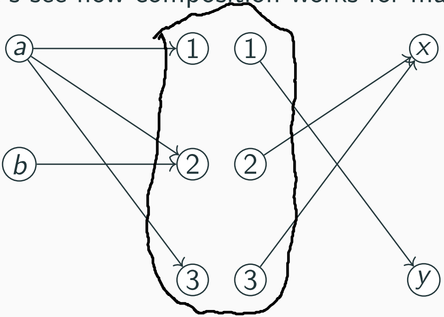
$$\begin{tikzcd}
1 & 2 \arrow[l] \\
4 \arrow[r] & 3 \arrow[u]
\end{tikzcd}$$

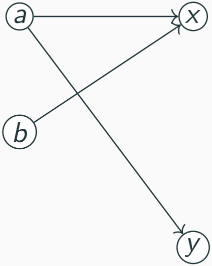
- The ordered pairs $R =$ $\{(2,1), (3,2), (4,3)\}$
- The matrix

$$\begin{array}{c c}
 & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \cdot & 1 & 0 & 0 \\ \cdot & \cdot & 1 & 0 \end{bmatrix}
\end{array}$$

- In words:

Now let's see how composition works for matrices representing relations.





$$R : \begin{array}{c} a \\ b \end{array}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad S : \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad S \circ R : \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Matrix representation of compositions

Let $A = \{a_1, \ldots, a_n\}$, $B = \{b_1, \ldots, b_m\}$ and $C = \{c_1, \ldots, c_p\}$.

The logical matrix M representing R is given by

$$M(i,j) = \begin{cases} 1 & \text{if} \quad (a_i, b_j) \in R \\ 0 & \text{if} \quad (a_i, b_j) \notin R \end{cases}$$

The logical matrix N representing S is given by

$$N(i,j) = \begin{cases} 1 & \text{if} \quad (b_i, c_j) \in S \\ 0 & \text{if} \quad (b_i, c_j) \notin S \end{cases}$$

Then the entries $P(i,j)$ of the *logical (Boolean) matrix product* $S \circ R$ are represented by matrix $P = MN$ given by

$$P(i,j) = \begin{cases} 1 & \text{if} \quad \text{there } \underline{\text{exists } l,\ 1 \leq l \leq m}, \text{ such that } \underline{M(i,l) = 1 \ \&\ N(l,j) = 1} \\ 0 & \text{if} \quad \text{otherwise.} \end{cases}$$

The example from before

Let R be the relation between $A = \{a, b\}$ and $B = \{1, 2, 3\}$ represented by

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

Similarly, let S be the relation between B and $C = \{x, y\}$ represented by

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

Then the matrix $P = MN$ representing $S \circ R$ is

$$P = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Detour: Boolean multiplication in Python

```
def booleanMM(m1, m2):
    # creating a zero matrix
    res = [ [0 for i in range(len(m2[0]))]
            for j in range(len(m1)) ]
    # computing the result
    for i in range(len(m1)):
        for j in range(len(m2[0])):
            for k in range(len(m2)):
                res[i][j] = (res[i][j] or
                    (m1[i][k] and m2[k][j]))
    return res

print(booleanMM([[0,0,1],[1,0,1]], [[1,0],[0,1],[0,0]]))
```

Detour: Boolean multiplication in Python numpy

```
import numpy as np
a = np.array([[0, 0, 1],
              [1, 0, 1]], dtype=bool)
b = np.array([[1, 0],
              [0, 1],
              [0, 0]], dtype=bool)

print(1*np.dot(a,b))
```

Travelling friends example: create and solve

$A = \{\text{Ann, Bob, Chris}\}$, $B = \{\text{France, Greece}\}$. Find $P = MN$ for $S \circ R$:

$R \subseteq A \times A$, $R(x, y)$ represents *"x is a friend of y"* with $M = \begin{array}{cc} & \begin{array}{ccc} a & b & c \end{array} \\ \begin{array}{c} a \\ b \\ c \end{array} & \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$

$S \subseteq A \times B$, $S(u, v)$ represents *"u visited v"* with $N = \begin{array}{cc} & \begin{array}{cc} f & g \end{array} \\ \begin{array}{c} a \\ b \\ c \end{array} & \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \end{array}$

$S \circ R$

$\circ$

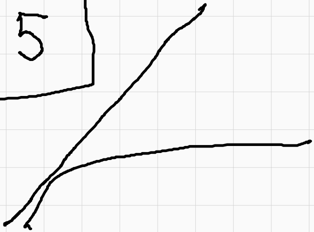
$$MN = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

DIY $\quad M = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} \quad MN - ?$

Thinking time: How would you decide if number 35 belongs to the list $\{1, 11, 22, 33, 44, 55\}$?

1 11 22 33 | 440 | 550667

$\boxed{35}$



$\log x$

x

DIY Another motivating example: ordering strings

Consider relations R, S and L on the set of all strings:

- L—Lexicographic ordering;
- uSv if, and only if, u is a Substring of v;
- uNv if, and only if, $len(u) \leq len(v)$.

Investigate which of those relations allow us to compare, order and search for **any** string. Use codes on the next page for some initial inspiration.

Lexicographic comparison code

```
print("apple" < "banana")
print("apple" == "Apple")
print("apple" < "Apple")
print(ord("a"))
print(ord("A"))
print("apple"<" apple")
print("apple"<"apple ")
```

Summary

Attendance code: 849377

- A relation R is represented by the *n by m matrix* $M(i,j)$ where
  $M(i,j) = \begin{cases} 1 & \text{if} \quad (a_i, b_j) \in R \\ 0 & \text{if} \quad (a_i, b_j) \notin R \end{cases}$
- Matrix product $P = MN$ is given by
  $P(i,j) = \begin{cases} 1 & \text{if} \quad \exists l,\ 1 \leq l \leq m, \text{ such that } M(i,l) = 1 \ \&\ N(l,j) = 1 \\ 0 & \text{if} \quad \text{otherwise.} \end{cases}$
- Ordering of elements is important.