School of Mathematics and Statistics
Te Kura Mātai Tatauranga

ENGR 123 **Tutorial 3**

1 For each relation on the given set, determine whether it is reflexive, symmetric, antisymmetric and/or transitive.

In each case where the relation doesn't have a property, give a short reason why or an example e.g. it isn't reflexive because ...

(a) On the set of people, $(a, b) \in R$ iff b is a maternal ancestor of a.

Solution: Anti-symmetric and transitive.
Not reflexive, either not your own mother, or not female, or ....
Not symmetric, your mother isn't your child.

(b) On the set of 2×2 matrices, a and b have the same determinant.

Solution: Reflexive, symmetric and transitive.
Not Anti-symmetric e.g. check I_2 and the 180°-rotation matrix.

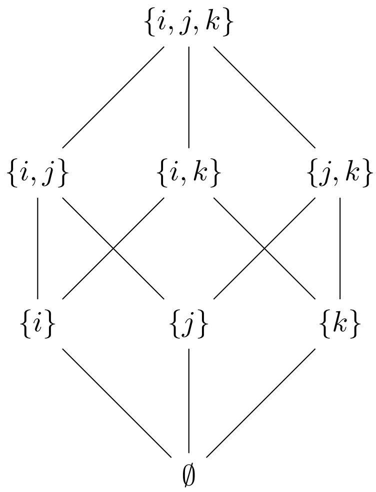
2 For the partition $\{0, 2\}, \{4\}, \{1, 3, 5\}$ of $\{0, 1, 2, 3, 4, 5\}$, list the ordered pairs in the equivalence relation produced by the partition.

Solution:

$$\Big\{(0,0),(0,2),(2,0),(2,2),$$
$$(4,4),$$
$$(1,1),(1,3),(1,5),(3,1),(3,3),(3,5),(5,1),(5,3),(5,5)\Big\}$$

3 Let $A = \mathcal{P}(\{j,k,l\})$ be the powerset (the set of all subsets) of $\{j,k,l\}$. Draw the Hasse diagram of the subset relation $\subset$ on A.

Solution:



4 Use induction to show that

$$\frac{1}{1} + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^n} = \frac{2^{n+1}-1}{2^n}$$

Solution:

Proof:

Base case ($n = 0$): Check both sides. LHS $= \frac{1}{1} = 1$ and RHS $= \frac{2^{0+1}-1}{2^0} = \frac{2-1}{1} = 1$

They agree!

Induction hypothesis ($n = k$): Assume that

$$\frac{1}{1} + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^k} = \frac{2^{k+1}-1}{2^k}$$

Inductive step ($n = k+1$): Working from the left hand side

$$\begin{aligned}
\frac{1}{1} + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^{k+1}} &= \frac{1}{1} + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^k} + \frac{1}{2^{k+1}} \\
&= \left(\frac{2^{k+1}-1}{2^k}\right) + \frac{1}{2^{k+1}} \\
&= \frac{2 \times 2^{k+1} - 2}{2^{k+1}} + \frac{1}{2^{k+1}} \\
&= \frac{2^{k+2}-1}{2^{k+1}}
\end{aligned}$$

which is the right-hand side, they match!

5 Consider the following pseudocode:

Algorithm 1: `X(A)`

```
   input: sequence A = (a_1, ..., a_n) of n distinct numbers, where n is a power of two
1  if (n = 1) then
2  |  return a_1
3  else
4  |  c = X(a_1, ..., a_{n/2})
5  |  d = X(a_{n/2+1}, ..., a_n)
6  |  if (c > d) then
7  |  |  return c
8  |  else
9  |  |  return d
```

(a) What does algorithm `X(A)` do?

Solution: `X(A)` finds the maximum element of a sequence $A = (a_1, \ldots, a_n)$.

(b) Let $C(n)$ be the number of comparisons `X()` performs on a list of length n (where $n = 2^k$ is a power of two). Write down a formula for $C(n)$ and prove your answer using induction.

Solution:

Theorem. $C(n) = n - 1$.

Proof. By induction on n.

Base case: If $n = 1$ then no comparisons were performed, so $C(1) = 0$.

Inductive hypothesis: Assume true for $n = 2^k$.

Inductive step: Check for $n = 2^{k+1}$.

$$\begin{aligned} C(2^{k+1}) &= \underbrace{C(2^k))}_{\text{line 4}} + \underbrace{C(2^k))}_{\text{line 5}} + \underbrace{1}_{\text{line 6}} \\ &= 2^k - 1 + 2^k - 1 + 1 \\ &= 2^{k+1} - 1 \end{aligned}$$

as required. QED.

(c) Is it possible to design a more efficient algorithm that performs the same task (that is, one which uses fewer comparisons)? Why or why not?

Solution. No, you cannot find the maximum element of a sequence without comparing all the elements.