EDAA40

Discrete Structures in Computer Science

– Exercises –

Jorn W. Janneck

2021.05.19

Table of Contents

Some exercises are labeled (*). These are exercises primarily intended for exam preparation toward the end of the course, because they may contain notation or a concept that is introduced later in the course (i.e. later than the segment they occur in). It does not mean that they are necessarily more difficult. If you want to try your hand on them earlier, see if you can look up the notation, or ask me.

In fact, this last bit is general advice: if you get stuck with anything, please let me know using one of the channels provided during the course.

1. Sets

1.1.

Using set builder notation, define the following sets (you can use the usual sets of numbers as starting points):

1. The set E of all even natural numbers (starting at 0).

   $E =$

2. Given the set $D = \{d \in \mathbb{N} : d \leq 9\}$, define the family of sets $\{L_d : d \in D\}$ such that L_d is the set of all natural numbers whose decimal representation ends with the digit d.

   $L_d =$

3. The set T of natural numbers that are the product of exactly three different primes.

   $T =$

4. The set X of pairs of natural numbers such that in each pair, the left-hand number is a square number, and the right-hand number is the cube of the same value that the left-hand number is the square of.

   $X =$

1.2.

Given $A = \{a, b, c, d, e, f\}$, what is

1. $\#(\{s \in \mathcal{P}(A) : \#(s) = 0\}) =$
2. $\#(\{s \in \mathcal{P}(A) : \#(s) = 1\}) =$
3. $\#(\{s \in \mathcal{P}(A) : \#(s) = 2\}) =$
4. $\#(\{s \in \mathcal{P}(A) : \#(s) = 3\}) =$
5. $\#(\{s \in \mathcal{P}(A) : \#(s) = 4\}) =$
6. $\#(\{s \in \mathcal{P}(A) : \#(s) = 5\}) =$
7. $\#(\{s \in \mathcal{P}(A) : \#(s) = 6\}) =$

1.3.

Given $A = \{3, 4, 5, 6, 7, 8\}$, suppose we define the following sets

$$B = \left\{ \frac{a-b}{a+b} : a, b \in A \right\}$$

$$C = \left\{ \frac{a}{b} : a, b \in A, a \geq b \right\}$$

Give the number of elements in these sets as follows:

1. $\#(B) =$_____
2. $\#(C) =$_____

1.4.

Given $A = \{a \in \mathbb{N}^+ : a \leq 6\}$, let us define the following sets:

$$B = \left\{\frac{a}{b} : a, b \in A\right\}$$
$$C = \left\{\frac{a}{b} : a, b \in A, b|a\right\}$$
$$D = \left\{\frac{a}{b} : a, b \in A, a|b\right\}$$

Here, $a|b$ means that a evenly divides b, i.e. that there is an integer k such that $ak = b$.

Give the number of elements in these sets as follows:

1. $\#(B) =$
2. $\#(C) =$
3. $\#(D) =$

1.5.

Given $A = \{1, 2, 3, 4, 5, 6, 7\}, B = \{3, 4, 5, 6, 7, 8\}$, $C = \{ab : a \in A, b \in B\}$
$D = \{ab : a \in A, b \in B, a > b\}$, and $E = \{ab : a \in A, b \in A \cap B\}$:

1. $\#(C) =$
2. $\#(D) =$
3. $\#(E) =$

1.6.

Given $A = \{a, b, c, d, e, f\}$:

1. $\#(\mathcal{P}(A)) =$
2. $\#\{s \in \mathcal{P}(A) : \{a, e\} \subseteq s\} =$
3. $\#\{s \in \mathcal{P}(A) : \{a, e\} \subset s\} =$

1.7.

In axiomatic set theory, all entities are sets, and that includes numbers, which are represented by sets that are constructed in some special way. A common technique for building up natural "numbers" is called "von Neumann construction", and it works as follows. First, the number 0 (zero) is represented by the empty set $\emptyset$.

Starting from that, given any "number" n, the set representing the next number (that is, the number one greater than the previous one), let us call it n^+, is defined as:

$$n^+ = n \cup \{n\}$$

1. Construct the sets representing first few natural numbers:

   0 = $\emptyset$

   1 =

   2 =

   3 =

2. How would one compare numbers in this representation? Suppose m and n are von-Neumann-constructed numbers as above:

   $m \leq n$ iff

3. Compute the maximum and the minimum of two von Neumann numbers:

   (a) $\max(m, n) =$

   (b) $\min(m, n) =$

4. (*) Let's add two von Neumann numbers (we use the fact that every von Neumann number is either zero, or the "next number" to another number just one smaller):

$$\text{plus}(m, n) = \begin{cases} & \text{if} \quad n = \emptyset \\ & \text{if} \quad n = k^+ \end{cases}$$

2. Relations

2.1.

Suppose $A = \{n \in \mathbb{N} : 1 \leq n \leq 10\}$ and a family of relations

$$R_i = \{(a, b) \in A \times A : \mod (b, a) = i\} \qquad \text{for any } i \in \mathbb{N},$$

with $\mod (b, a)$ the remainder when dividing positive integer b by positive integer a.
So, for example, $R_3 = \{(a, b) \in A \times A : \mod (b, a) = 3\}$.

1. $\#R_4 =$

2. $\#R_5 =$

3. $\#R_0 =$

5. $\#R_{10} =$

6. $R_3(7) =$

7. $R_1(2) =$

8. $R_1(A) =$

9. $R_3(A) =$

2.2.

With $A = \{n \in \mathbb{N}^+ : n \leq 20\}$ and $R = \{(a, b) \in A^2 : a|b\}$ compute the following images of R:
($a|b$ means that a evenly divides b, i.e. that there is an integer k such that $ak = b$.)

1. $R(6) =$

2. $R(7) =$

3. $R(2) =$

4. $R(\{2, 5\}) =$

2.3.

With $A = \{n \in \mathbb{N}^+ : n \leq 10\}$ and $R = \{(a, (a + b)) : a \in A, b \in A, a \perp b\}$ compute the following images of R:
($a \perp b$ means that a and b are *coprime*, i.e. they their only common positive divisor is 1.)

1. $R(2) =$

2. $R(3) =$

3. $R(4) =$

4. $R(\{5, 7\}) =$

2.4.

With $A = \{1, 2, 3, 4, 5, 6, 7\}$, < the usual arithmetic order on A, and the relations $B = \{(a, b) \in A^2 : a < b\}$, and $C = \{(a, b) \in A^2 : a \perp b\}$ (the complement is supposed to be built with respect to $A \times A$):
($a \perp b$ means that a and b are *coprime,* i.e. they their only common positive divisor is 1.)

1. $\#(B) =$
2. $\#(C) =$
3. $\#(\overline{B}) =$
4. $\#(B^{-1}) =$

2.5.

With $A = \{2, 3, 4, 5, 6, 7\}$, $R = \{(a, b) \in A^2 : a \perp b\}$, and $S = \{(a, b) \in A^2 : a|b\}$. We are looking at the composition $S \circ R$ in this exercise.
($a \perp b$ means that a and b are *coprime,* i.e. they their only common positive divisor is 1. $a|b$ means that a evenly divides b, i.e. that there is an integer k such that $ak = b$.)

1. $\#(S \circ R) =$_____
2. $S \circ R(2) =$_____
3. $S \circ R(3) =$_____
4. $S \circ R(6) =$_____
5. $S \circ R(7) =$_____
6. $S \circ R$ is ... (circle those that apply)
   - … reflexive on A TRUE FALSE
   - … symmetric TRUE FALSE
   - … transitive TRUE FALSE
   - … antisymmetric TRUE FALSE
7. R is ... (circle those that apply)
   - … reflexive on A TRUE FALSE
   - … symmetric TRUE FALSE
   - … transitive TRUE FALSE
   - … antisymmetric TRUE FALSE

2.6.

With $A = \{1, 2, 3, 4, 5, 6, 7\}$, $<$ the usual arithmetic order on A, and $< \circ <$ its composition with itself:

1. $< (1) =$
2. $< \circ < (1) =$
3. $\{(a, b) \in A^2 : a(< \circ <)b\} =$

4. Is $< \circ <$ transitive?	YES	NO
5. Is $< \circ <$ reflexive on $A \times A$?	YES	NO

2.7.

With $A = [1, 7]$ in the **real numbers** $\mathbb{R}$, $<$ the usual arithmetic order on A, and $< \circ <$ its composition with itself:

1. $< (1) =$
2. $< \circ < (1) =$

2.8.

Suppose $A = \{n \in \mathbb{N} : 1 \leq n \leq 10\}$ and a relation

$$R_3 = \{(a, b) \in A \times A : \mod (b, a) = 3\}$$

with $\mod (b, a)$ the remainder when dividing positive integer b by positive integer a.

Let $T = R_3 \circ R_3^{-1}$.

1. $\#T =$
2. $T(1) =$
3. $T(7) =$
4. $T(A) =$
5. T is ... (circle those that apply)

… reflexive on A	TRUE	FALSE
… symmetric	TRUE	FALSE
… transitive	TRUE	FALSE
… antisymmetric	TRUE	FALSE

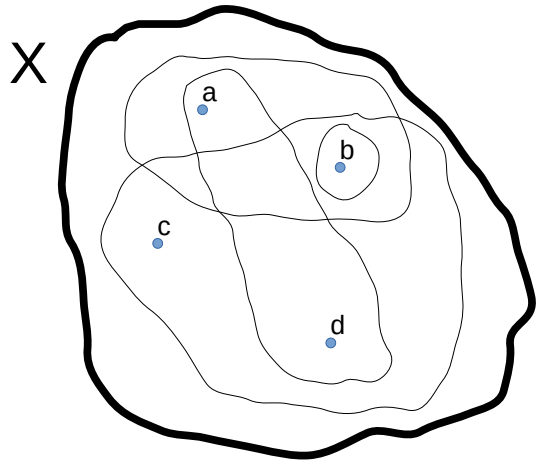


Figure 1: A set X and a few non-empty subsets.

2.9. (*)

Suppose a non-empty finite set X and a set $E \subseteq \mathcal{P}(X) \setminus \{\emptyset\}$ of non-empty subsets of X.[1] For example, in Fig. 1, $X = \{a, b, c, d\}$ and $E = \{\{a, b\}, \{a, d\}, \{b\}, \{b, c, d\}\}$.

Now consider the relation $R_E \subseteq X \times X$, defined as

$$R_E = \{(v, w) : (v \neq w) \wedge \exists e \in E \ (\{v, w\} \subseteq e)\}$$

Such a relation is called the *2-section* of E.

List the elements of 2-section R_E for the definitions in the example above, shown also in Fig. 1:

$R_E =$

1 A structure like this – a set X and a set E of non-empty subsets of X – is also called a *hypergraph*, another discrete structure, with X being the vertices and E the *hyperedges* between them (note that each hyperedge connects any positive number of vertices). It has applications, for instance, in describing the structure of relational databases, where the elements of X are the column headers and each element of E represents a table. Such a hypergraph can then be used to investigate the structure of a database, e.g. for the purpose of normalizing it.

2.10. (*)

As in exercise 2.9, given non-empty finite set X and a set $E \subseteq \mathcal{P}(X) \setminus \{\emptyset\}$ of non-empty subsets of X, we define the 2-section $R_E \subseteq X \times X$ as a relation on X with

$$R_E = \{(v, w) : (v \neq w) \wedge \exists e \in E \ (\{v, w\} \subseteq e)\}$$

This exercise is about the properties of the relation R_E for all possible X and E.

In the following table, tick one box in each row, depending on whether R_E **always** has the property in the left column (for every X and E as above), whether R_E **sometimes** has that property (which means there is at least one pair of X and E as above for which is has that property, and at least one for which it does not), or whether it **never** has the property, for no X and E as above.

Hint: Do consider "corner cases", such as what happens when X has only one element or E has none (note that while the elements of E must be non-empty sets, E itself *can* be empty), etc.

	always	**sometimes**	**never**
reflexive over X			
transitive			
symmetric			
antisymmetric			
asymmetric			

2.11. (*)

As in exercise 2.9, given non-empty finite set X and a set $E \subseteq \mathcal{P}(X) \setminus \{\emptyset\}$ of non-empty subsets of X, we define the 2-section $R_E \subseteq X \times X$ as a relation on X with

$$R_E = \{(v, w) : (v \neq w) \wedge \exists e \in E \ (\{v, w\} \subseteq e)\}$$

Given an X and an E, this relation R_E is uniquely defined, i.e. there are not two different such relations for the same X and E. This exercise is about whether the reverse is also true.

1. For all non-empty sets X, and all pairs of distinct sets $E_1, E_2 \subseteq \mathcal{P}(X) \setminus \{\emptyset\}$ of non-empty subsets of X, with $E_1 \neq E_2$, is it the case that the two 2-sections R_{E_1} and R_{E_2} are always different, i.e. that $R_{E_1} \neq R_{E_2}$ whenever $E_1 \neq E_2$ (circle the right answer)?

   yes no

2. If **yes**, prove it. This means you need to use the definition of a 2-section, and show that it implies that two different $E_1, E_2 \subseteq \mathcal{P}(X) \setminus \emptyset$ will always result in two different 2-sections R_{E_1} and R_{E_2}. Note that two sets are different iff there is at least one element that is in one but not the other.

   If **no**, provide a counterexample. That means you need to give a non-empty X, and then two-different sets of (non-empty) subsets of X, E_1 and E_2 that result the same 2-section, i.e. $R_{E_1} = R_{E_2}$. In that case, also provide the common 2-section of E_1 and E_2.

2.12. (*)

In addition to a non-empty finite set X and a set $E \subseteq \mathcal{P}(X) \setminus \{\emptyset\}$ of non-empty subsets of X, as in exercise 2.9, suppose a set Y (disjoint from X, so $X \cap Y = \emptyset$) and a bijection $f : E \longleftrightarrow Y$. Basically, you can think of Y as a set of labels, and f as a function that assigns each element of E uniquely one of those labels.

Now consider a relation $Q_E \in X \times Y$ such that for any $v \in X$ and $w \in Y$, it is the case that

$(v, w) \in Q_E$ iff v is an element of the $e \in E$ that f maps to w.

1. For the example from task 2.9 shown in Fig. 1, suppose that $Y = \{p, q, r, s\}$ and $f = \{(\{a, b\}, p), (\{a, d\}, q), (\{b\}, r), (\{b, c, d\}, s)\}$. List the elements of Q_E in that case.

   $Q_E =$

2. Now give a **general** (not just for the example) **formal** (no words, just math) definition of the relation $Q_E \subseteq X \times Y$, for any non-empty finite X, a set $E \subseteq \mathcal{P}(X) \setminus \{\emptyset\}$, a set Y, and a bijection $f : E \longleftrightarrow Y$:

   $Q_E =$

3. Functions

3.1.

Define two sets A and B, as well as a function $f : A \longrightarrow B$, such that f is **surjective** and **not injective**.

$A =$

$B =$

$f =$

3.2.

Suppose $f : x \mapsto (x + 1)^2$ with domain A and codomain B.

1. If $A = [-2, 2]$, what is B so that f is surjective? $B =$
2. If $B = [0, 4]$, what is a set A such that f is bijective? $A =$
3. If $B = [0, 4]$, what is the largest possible set A? $A =$

3.3.

Assume you have a surjection $s : A \twoheadrightarrow B$ and an injection $j : B \hookrightarrow C$.

1. Their composition $j \circ s$ is not always injective. Show this using a counterexample, by giving definitions for A, B, C, s, and j, and demonstrating that their composition $j \circ s$ is not injective.

   $A =$

   $B =$

   $C =$

   $s =$

   $j =$

   Now demonstrate that the composition $j \circ s$ is not injective:

2. Sometimes, though, their composition $j \circ s$ *is* injective. Show this by giving definitions for A, B, C, s, and j, such that the composition $j \circ s$ is injective. (You do not need to prove that it is injective, it suffices that it is the case.)

   $A =$

   $B =$

   $C =$

   $s =$

   $j =$

3. What property (in addition to being surjective as required) does s have to have, so that $j \circ s$ is injective?

4. Their composition $j \circ s$ is also not always surjective. Show this using a counterexample, by giving definitions for A, B, C, s, and j, and demonstrating that their composition $j \circ s$ is not surjective.

   $A =$

   $B =$

   $C =$

   $s =$

   $j =$

   Now demonstrate that the composition $j \circ s$ is not surjective:

3.4.

Assume you have an injection $j : A \hookrightarrow B$ and a surjection $s : B \twoheadrightarrow C$.

1. Their composition is not always injective. Show this using a counterexample, by giving definitions for A, B, C, j, and s, and demonstrating that their composition $s \circ j$ is not injective.

   $A =$

   $B =$

   $C =$

   $j =$

   $s =$

   Now demonstrate that the composition $s \circ j$ is not injective:

2. Their composition is not always surjective. Show this using a counterexample, by giving definitions for A, B, C, j, and s, and demonstrating that their composition $s \circ j$ is not surjective.

   $A =$

   $B =$

   $C =$

   $j =$

   $s =$

   Now demonstrate that the composition $s \circ j$ is not surjective:

3.5. (*)

Remember that if we have two injections $f : A \hookrightarrow B$ and $g : B \hookrightarrow C$, we know that their composition $g \circ f$ is always injective.

However, suppose we have an injection $h : A \hookrightarrow C$ which we know to be the result of composing $f : A \longrightarrow B$ and $g : B \longrightarrow C$, i.e. $h = g \circ f$, but we don't know those two functions, nor do we know B. Knowing that h is injective, what can we say about f, g, and B?

In the following, you will be given a few statements. You are asked to **decide whether they follow from the fact that h is injective**. If you think a statement does follow from that, prove it. Otherwise, show a counterexample (a counterexample consists of definitions of A, B, C, and f, g, and h, such that the h is injective, $h = g \circ f$, and the statement is not true).

1. f must be injective (circle correct answer). YES NO
   If yes, prove it, if no, find a counterexample.

2. g must be injective (circle correct answer). YES NO
   If yes, prove it, if no, find a counterexample.

3.6.

Suppose you have **injections** $f : A \hookrightarrow B$ and $g : A \hookrightarrow B$, as well as a **non-empty** set $S \subset A$ (note that S is a **proper** subset of A). Now let's define a function $h : A \longrightarrow B$ as follows:

$$h : x \mapsto \begin{cases} f(x) & \text{for } x \in S \\ g(x) & \text{for } x \notin S \end{cases}$$

This function is not, in general, injective.
Whether it is injective depends on the definitions of A, B, f, g, and S.

1. Give definitions for A, B, f, g, and S such that the h above is injective.

   A =

   B =

   f =

   g =

   S =

2. Give definitions for A, B, f, g, and S such that the h above is **not** injective.

   A =

   B =

   f =

   g =

   S =

3. Give a general formal criterion, depending only on A, B, f, g, and S (not necessarily all of them), that defines the condition under which h is injective. (Hint: Remember, f and g are already injective.)

   h is injective if and only if

   Note: You are **not** supposed to reiterate the definition of injectivity for h, but rather give an expression involving at most A, B, f, g, and S (but **not** h) that is true if an only if they lead to an injective h.

3.7.

Suppose we have a set A that is **totally ordered** by a strict total order relation $<$ on A. A function $f : A \longrightarrow A$ is called *strictly monotonic* iff for any $a, b \in A$ it is the case that $a < b$ implies that $f(a) < f(b)$.

1. Show that such a strictly monotonic function f is always injective.

2. A strictly monotonic function is not, however, necessarily surjective. Demonstrate this by giving a counterexample.

3.8.

Define **one** set A, as well as a function $f : A \longrightarrow A$, such that f is **surjective** and **not injective**.

$A =$

$f :$

3.9.

In the following table, determine the set of numbers that the expression in the left column evaluates to, and write it out using enumerations, set-builder notation, and basic sets (such as $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}^+$, etc.).

Unless otherwise specified, intervals are intervals of real numbers, i.e. in $\mathbb{R}$.

These are the functions that occur in the table:

$$\begin{aligned} \text{incr} : \mathbb{R} &\longrightarrow \mathbb{R} \\ r &\mapsto r + 1 \end{aligned}$$

$$\begin{aligned} \text{sqrt} : \mathbb{R}^+ &\longrightarrow \mathbb{R}^+ \\ r &\mapsto +\sqrt{r} \end{aligned}$$

$$\begin{aligned} \text{inv} : \mathbb{R} \setminus \{0\} &\longrightarrow \mathbb{R} \\ r &\mapsto \frac{1}{r} \end{aligned}$$

expression...	**... evaluates to**
$\{x \in \mathbb{Z} : x > 4 \wedge x < 2\}$	$\emptyset$
$\mathrm{incr}(\{1, 2, 3, 4\})$	$\{2, 3, 4, 5\}$
$\ln(\mathbb{R}^+)$	$[1, +\infty[$ or $\{r \in \mathbb{R} : r \geq 1\}$ or $\mathbb{R}^+ \setminus [0, 1[$ etc.
$\mathrm{sqrt}(]1, 4])$	$]1, 2]$
$\bigcap_{i \in \mathbb{N}^+} \left]-\frac{1}{i}, \frac{1}{i}\right]$	
$\mathrm{incr}[\{0\}]$	
$\bigcup_{i \in \mathbb{N}^+} \left[\frac{1}{i}, 1\right]$	
$\mathrm{incr}(]0, 1])$	
$\mathrm{incr}[]0, 1]]$	
$\bigcap_{i \in \mathbb{N}^+} \left[0, \frac{1}{i}\right]$	
$\bigcap_{i \in \mathbb{N}^+} \left]0, \frac{1}{i}\right]$	
$\bigcap_{i \in \mathbb{N}^+} \left[\frac{-1}{i}, \frac{1}{i}\right]$	
$\bigcap_{i \in \mathbb{N}^+} \left]\frac{-1}{i}, 1\right]$	
$\mathrm{sqrt}([0, 0.1])$	
$\mathrm{sqrt}[[0, 0.1]]$	
$\mathrm{sqrt}[]0, 0.1]]$	
$\bigcup_{i \in \mathbb{N}^+} \left]\frac{1}{i+1}, \frac{1}{i}\right]$	
$\bigcup_{i \in \mathbb{N}^+} \left]\frac{1}{i+1}, \frac{1}{i}\right[$	
$\mathrm{inv}(]0, 0.1])$	
$\mathrm{inv}[]0, 0.1]]$	

4. Infinity

4.1.

Recall that a set is infinite iff it is equinumerous to a proper subset of itself. Show that the set of all non-negative rational numbers (i.e. including 0) $\mathbb{Q}_0^+$ is infinite in two steps:

1. Define a **proper** subset of $A \subset \mathbb{Q}_0^+$ – to make this a little more interesting, this subset should be wholly contained between two rational numbers $a, b \in \mathbb{Q}$, i.e. for all $r \in A$, it should be true that $a \leq r \leq b$. (That does not mean that $A = [a, b]_{\mathbb{Q}}$, only that $A \subseteq [a, b]_{\mathbb{Q}}$ for some $a, b \in \mathbb{Q}$.)

   $A =$

2. Now construct a **bijection** $f : \mathbb{Q}_0^+ \longleftrightarrow A$

   $f : q \mapsto$

Note: You do **not** need to *prove* that your f is bijective. For this task it is sufficient if the function you specify has that property.

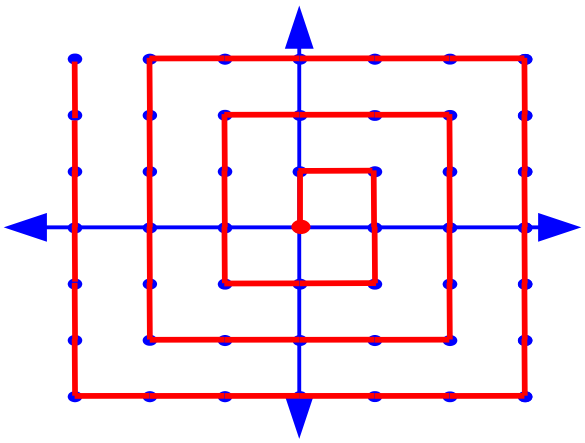


Figure 2: Graphical sketch of a bijection between the natural numbers and pairs of integers.

4.2.

Show that $\mathbb{Z}^2$ is equinumerous to $\mathbb{N}$ by demonstrating that there exists a bijection between the two sets. Fig. 2 shows such a bijection.

However, actually constructing is quite cumbersome. Instead, according to CSB, it suffices to construct two injections $j_1 : \mathbb{N} \hookrightarrow \mathbb{Z}^2$ and $j_2 : \mathbb{Z}^2 \hookrightarrow \mathbb{N}$ to demonstrate the existince of a bijection.

Define two such injections.

4.3.

In the lecture on infinity, it is said that $\#(\mathbb{N}) \leq \#(2^{\mathbb{N}})$. Show this.

Hint: You need to define something in order to prove this, and then show that the thing you defined has a certain property. Go step by step:

1. Start by writing down what you need to define and what property it must have.
2. Then define it.
3. Then show that it has the property.

4.4.

Show that $[0,1] \sim]0,1[$ in the real numbers, i.e. that the open interval from 0 to 1 is equinumerous to the closed one from 0 to 1.

1. Show this using CSB.

2. Show this without using CSB, by constructing an actual bijection $b : [0,1] \longleftrightarrow]0,1[$ between the two intervals.

4.5.

In the lecture on infinity, it is claimed that for any set A, we have $\#A < \#\mathcal{P}(A)$, i.e. a power set is always strictly larger than the set it is the power set of. (This then implies the existence of infinitely many transfinite cardinal numbers, because you can apply the power set operation over and over again.)

Show that $\#A < \#\mathcal{P}(A)$ as follows. (If you get stuck, do let me know.)

The path here is via a slight generalization of Cantor's diagonal proof. Basically, you need to show that for any set A, there is no surjective function $f : A \twoheadrightarrow \mathcal{P}(A)$.

In the diagonal proof in the lecture, Cantor assumed that we have such a surjective function from $\mathbb{N}$ to $\{0,1\}^{\mathbb{N}}$, and then constructs an element of the codomain of that function that it cannot possibly map to – which then contradicts the assumption that it was surjective and thus implies that there cannot be a surjective function from $\mathbb{N}$ to $\{0,1\}^{\mathbb{N}}$.

In the diagonal proof as shown in the lecture, the domain of the function was $\mathbb{N}$, and the codomain $\{0,1\}^{\mathbb{N}}$, the infinite sequences of 0s and 1s, each of which is itself really a function $s : \mathbb{N} \longrightarrow \{0,1\}$, which is why the codomain of f is $\{0,1\}^{\mathbb{N}}$, or equivalently $2^{\mathbb{N}}$.

In the lecture, it is said that the set $\{0,1\}^{\mathbb{N}}$ of infinite sequences of 0s and 1s can also be thought of as the powerset $\mathcal{P}(\mathbb{N})$ of the natural numbers. This is because you could represent an infinite sequence of 0s and 1s $s : \mathbb{N} \longrightarrow \{0,1\}$ simply as the set of all natural numbers for which s is 1, i.e. $\{k \in \mathbb{N} : s(k) = 1\}$. Conversely, for any set $A \subseteq \mathbb{N}$ you could construct a sequence function $s : \mathbb{N} \longrightarrow \{0,1\}$ such that

$$s(k) = \begin{cases} 1 & \text{for } s \in A \\ 0 & \text{for } s \notin A \end{cases}.$$

In other words, there is a 1:1 correspondence between the two. But this means that you could think of Cantor's diagonal proof already as a special case of what we want to show above, namely $\#\mathbb{N} < \#\mathcal{P}(\mathbb{N})$.

Armed with this insight, you now need to generalize this just a tiny bit. Suppose there is a function

$$f : A \longrightarrow \mathcal{P}(A)$$

You need to use it to build a set $\overline{D} \in \mathcal{P}(A)$ (or simply $\overline{D} \subseteq A$) such that $\overline{D} \notin \text{range } f$. If such a $\overline{D}$ exists for any f, it means there cannot be a surjective f and thus there cannot be a bijection. In the proof in the lecture, we arrived at this $\overline{D}$ by first constructing the diagonal sequence D from f, and then inverting it. We shall do the same now here for any set A.

1. First, define a set $D \in \mathcal{P}(A)$ (or $D \subseteq A$) that corresponds to the interpretation of the diagonal sequence D from the lecture as a set.

   $D = \{a \in A : \_\_\_\_\_\}$

2. Now define $\overline{D}$, corresponding to the inverse of the diagonal sequence from the proof as shown in the lecture:

   $\overline{D} = \{a \in A : \_\_\_\_\_\}$

3. All that is left is to show is that given any function f, the corresponding $\overline{D} \notin \text{range } f$.

   You can show this by demonstrating that for any $a \in A$, $f(a) \neq \overline{D}$. Remember that two sets are different iff there is at least one element that is in one of them, but not the other:

   $f(a) \neq \overline{D}$ because…

5. Induction and recursion

5.1.

Show by simple induction that for every positive integer n, $5^n - 1$ is divisible by 4.
(SLAM Exercise 4.1)

Basis, n = 1:

Induction step from n to n+1:

5.2.

Let us use $\mathbb{P}$ as the name for the set of all prime numbers, that is positive integers greater than 1 that are only divisible by 1 and themselves, so $\mathbb{P} = \{2, 3, 5, 7, 11, 13, ...\}$. You can use $\mathbb{P}$ in answering the following questions, and also the "divides" relation, defined as $a|b$ iff $\exists k(k \in \mathbb{N}^+ \wedge ka = b)$.

1. The number *n primorial* is the product of all prime numbers less than or equal to n, i.e. $\prod_{k \in \{p \in \mathbb{P} : p \leq n\}} k$. Let us call the function that computes n primorial P, so for example, $P(3) = 2 \cdot 3 = 6$, $P(4) = 2 \cdot 3 = 6$, $P(5) = 2 \cdot 3 \cdot 5 = 30$, $P(6) = 2 \cdot 3 \cdot 5 = 30$, $P(7) = 2 \cdot 3 \cdot 5 \cdot 7 = 210$ and so forth. The first primorial number is defined to be $P(1) = 1$.
   Give a recursive definition of the function $P : \mathbb{N}^+ \longrightarrow \mathbb{N}^+$ computing n primorial for any $n \in \mathbb{N}^+$, as follows:

$$P : n \mapsto \begin{cases} 1 & \text{for } n = 1 \\ \underline{\qquad\qquad\qquad} & \text{for } n > 1, n \notin \mathbb{P} \\ \underline{\qquad\qquad\qquad} & \text{for } n > 1, n \in \mathbb{P} \end{cases}$$

2. Is the function $P : \mathbb{N}^+ \longrightarrow \mathbb{N}^+$... (circle the answer)

   (a) injective? YES NO

   (b) surjective? YES NO

3. Recursively define an **injective function** $Q : \mathbb{N}^+ \hookrightarrow \mathbb{P}$.
   Use the fact that for any $k \in \mathbb{N}^+$, the number $P(k) + 1$ is a prime number (a so-called *primorial prime*).
   Hint: It's NOT as simple as mapping n to $P(n) + 1$. (Make sure you understand why that is)

$$Q : n \mapsto \begin{cases} P(1) + 1 & \text{for } n = 1 \\ \underline{\qquad\qquad\qquad\qquad\qquad} & \text{for } n > 1 \end{cases}$$

4. Prove that Q above is injective.
   You may use the fact that $n \leq P(n)$ for all $n \in \mathbb{N}^+$ without needing to prove it.

   Hint: Answering this might become easier if you use a result from a previous task.

5.3.

Given a set A, the set A^* is the set of all finite strings in A, including the empty string, ε.

1. Define a recursive function $\text{len} : A^* \longrightarrow \mathbb{N}$ that for each string $s \in A^*$ computes its length, i.e. the number of symbols in it.

   $\text{len} : A^* \longrightarrow \mathbb{N}$

   $s \mapsto$

2. In order to show that the function as defined above is well-defined, we need to find a well-founded ordering $\prec$ of its domain, A^*, such that a recursive call calls the function only on strictly "smaller" (according to that order) elements.

   Define such an ordering between two strings $a_1a_2...a_n$ and $b_1b_2...b_m$ of length n and m, respectively:

   $a_1a_2...a_n \prec b_1b_2...b_m \qquad \Longleftrightarrow$

5.4.

Given a set A, the set A^* is the set of all finite strings in A, including the empty string, ε.

Define a recursive function $\mathrm{count} : A \times A^* \longrightarrow \mathbb{N}$ that for each symbol $a \in A$ and each string $s \in A^*$ computes the number of occurrences of that symbol in the string, e.g.

$$\mathrm{count}(4, 12345654321) = 2$$

$$\mathrm{count} : A \times A^* \longrightarrow \mathbb{N}$$

$$a, s \mapsto$$

5.5.

Given a set A, the set A^* is the set of all finite strings in A, including the empty string, ε.

Define a recursive function $\mathrm{set} : A^* \longrightarrow \mathcal{P}(A)$ that computes for each string $s \in A^*$ the set of all symbols occurring in it at least once:

$$\mathrm{set} : A^* \longrightarrow \mathcal{P}(A)$$

$$s \mapsto$$

5.6.

Recall that $\{0,1\}^*$ is the set of all finite sequences of 0s and 1s. Define an **injective** function $f : \{0,1\}^* \hookrightarrow \mathbb{N}$.

Keep in mind that sequences of 0s and 1s may start and end with any number of 0s, so interpreting the string simply as a binary number is not going to result in an injective function, because 00100100 and 100100 would be mapped to the same natural number.

Use recursion over the structure of the sequence, as follows. The first case deals with the empty sequence, the other two cases "peel off" the first element in the sequence and the rest of the sequence is called s'.

$$f : s \mapsto \begin{cases} \_\_\_\_\_ & \text{for } s = \varepsilon \\ \_\_\_\_\_ & \text{for } s = 0s', s' \in \{0,1\}^* \\ \_\_\_\_\_ & \text{for } s = 1s', s' \in \{0,1\}^* \end{cases}$$

5.7.

As above, $\{0,1\}^*$ is the set of all finite sequences of 0s and 1s. Define an **injective** function $g : \mathbb{N} \hookrightarrow \{0,1\}^*$.

Use recursion over $\mathbb{N}$, as follows. The first case deals with 0, the other with positive numbers, where you can use the values for smaller numbers.

$$g : n \mapsto \begin{cases} \_\_\_\_\_ & \text{for } n = 0 \\ \_\_\_\_\_ & \text{for } n > 0 \end{cases}$$

5.8.

Let $A = \{a, b, c\}$ and $X = \{x, y\}$, and correspondingly A^* and X^* be the sets of finite sequences in A and X, respectively.

Using recursion over the structure of the sequence, define two **injections** $f : X^* \hookrightarrow A^*$ and $g : A^* \hookrightarrow X^*$, as follows. In both definitions, the first case deals with the empty sequence, the other cases “peel off” the first element in the sequence and the rest of the sequence is called s'.

$$f : s \mapsto \begin{cases} \_\_\_\_\_ & \text{for } s = \varepsilon \\ \_\_\_\_\_ & \text{for } s = xs', s' \in X^* \\ \_\_\_\_\_ & \text{for } s = ys', s' \in X^* \end{cases}$$

$$g : s \mapsto \begin{cases} \_\_\_\_\_ & \text{for } s = \varepsilon \\ \_\_\_\_\_ & \text{for } s = as', s' \in A^* \\ \_\_\_\_\_ & \text{for } s = bs', s' \in A^* \\ \_\_\_\_\_ & \text{for } s = cs', s' \in A^* \end{cases}$$

5.9.

Consider the lower-case alphabet $A = \{"a", ..., "z"\}$ and the set
$C = A \cup \{"(", ")", "\neg", " \vee ", " \wedge "\}$ of characters.
We define a small language $\mathcal{L} \subseteq C^*$ of propositional formulae over the set of variable names $V = A^* \setminus \{\varepsilon\}$, and the following set of rules $R = \{R_1, R_2, R_3\}$ with

$$R_1 = \{(s, "\neg" s) : s \in C^*\}$$
$$R_2 = \{(s_1, s_2, "(" s_1 " \vee " s_2 ")") : s_1, s_2 \in C^*\}$$
$$R_3 = \{(s_1, s_2, "(" s_1 " \wedge " s_2 ")") : s_1, s_2 \in C^*\}$$

such that $\mathcal{L} = R[V]$.

1. Show that $\mathcal{L} \subset C^*$ by giving a string $s \in C^*$ such that $s \notin \mathcal{L}$:

   $s =$

2. Give three strings $s_1, s_2, s_3 \in C^* \setminus \mathcal{L}$ such that $(s_1, s_2, s_3) \in R_3$:

   $s_1 =$

   $s_2 =$

   $s_3 =$

3. Assume a function $E : V \longrightarrow \{0, 1\}$ that assigns every variable name a value in $\{0, 1\}$. Using **structural recursion**, define an evaluation function $\mathrm{eval}_E : \mathcal{L} \longrightarrow \{0, 1\}$ that interprets the formulae in $\mathcal{L}$ in a way consistent with the usual interpretation of the symbols $\neg$ (not), $\vee$ (or), and $\wedge$ (and) in propositional logic. **Use arithmetic operators (+, -, *, min, max) to compute with the values 0 and 1.**

$$\mathrm{eval}_E : s \mapsto \begin{cases} \_\_\_\_\_ & \text{for } s \in V \\ \_\_\_\_\_ & \text{for } s = \neg s' \\ \_\_\_\_\_ & \text{for } s_1, s_2 \in \mathcal{L}, s = (s_1 \vee s_2) \\ \_\_\_\_\_ & \text{for } s_1, s_2 \in \mathcal{L}, s = (s_1 \wedge s_2) \end{cases}$$

5.10.

Suppose we have a set of five characters $C = \{"a", "b", "(", ")", ","\}$, the set $S = \{"a", "b"\}$ consisting only of the letters a and b, and a relation $R = \{(s_1, s_2, "(" s_1 ", " s_2 ")") : s_1, s_2 \in C^*\}$.

As you can see, R is a 3-place relation. Let us define a function $F : \mathcal{P}(C^*) \longrightarrow \mathcal{P}(C^*)$ such that

$$F : X \mapsto R(X \times X)$$

In other words, $F(X) = \{y : x_1, x_2 \in X, (x_1, x_2, y) \in R\}$.

Now let $F^n(X)$ be the set that results from applying F to some set $X \subseteq C^*$ n times in a row, with $F^0(X) = X$, $F^1(X) = F(X)$, $F^2 = F(F(X))$ and so forth, so that $F^{n+1}(X) = F(F^n(X))$.

1. Give an element in $F^0(S)$:

2. Give an element in $F^1(S)$:

3. Give an element in $F^2(S)$:

4. Suppose $U = \bigcup_{n \in \mathbb{N}} F^n(S)$.

   Give an element in $R[S] \setminus U$, or write “none” if no such element exists:

5.11.

Suppose we have a set A partially ordered by a strict order $<$. What would you need to show in order to demonstrate that $(A, <)$ is **not** well-founded?

5.12.

1. Is the set $\mathcal{P}(\mathbb{Q})$ under strict set inclusion $\subset$ well-founded? (circle answer)

   YES NO

2. Prove it or provide a counterexample.

3. Is the set $\mathcal{P}(\mathbb{N})$ under strict set inclusion $\subset$ well-founded? (circle answer)

   YES NO

4. Prove it or provide a counterexample.

5.13.

Let the set $\mathbb{S}$ be the set of all finite strings consisting of lower-case characters from a to z, including the empty string “”. So, for example, “abc”, “string”, and “ordered” are all members of $\mathbb{S}$.
The *length* of a string is the number of characters in it. We shall use $\mathrm{len}(s)$ for the number of characters in s. Note that $\mathrm{len}(\text{“”}) = 0$.

Let < be the **strict prefix order** on $\mathbb{S}$. This means that for any two strings s and t, we have $s < t$ if and only if s is a proper prefix of t, i.e. t starts with s and then contains at least one more character. For example, $\text{“}abc\text{”} < \text{“}abcd\text{”}$ and $\text{“”} < \text{“}xyz\text{”}$, but $\text{“}abc\text{”} \not< \text{“}abc\text{”}$, $\text{“}ab\text{”} \not< \text{“}xyz\text{”}$ and of course $\text{“}xyz\text{”} \not< \text{“”}$.

Note that for any two strings s and t, it is always the case that $s < t \Rightarrow \mathrm{len}(s) < \mathrm{len}(t)$.

1. $(\mathbb{S}, <)$ is partially / totally ordered (circle the one that applies).

   (For the purposes of this question, “partially” should be understood as the opposite of “totally”, rather than as its generalization.)

2. Is $(\mathbb{S}, <)$ well-founded? (circle answer)

   YES NO

6. Trees and graphs

6.1.

Recall that a *directed graph* (V, E) is defined as a finite set V of vertices and a relation $E \subseteq V \times V$ between them.

This question is about the properties of that relation. In the table below, make one mark in each row for the property in the left column, depending on whether all, some, or no relations defining a graph have that property. Put the mark in the corresponding ALL box, if **all relations** defining a graph have the corresponding property, the NONE box, if **no relation** has it, and the SOME box if at least one relation does, and at least one does not.

	ALL	SOME	NONE
reflexive over V			
transitive			
symmetric			
antisymmetric			
asymmetric			

6.2.

Recall that a *rooted tree* is a graph (T, R) such that, if the set T of nodes is not empty, then there is a node $a \in T$ (the root) such that for every $x \in T$ with $x \neq a$ there is exactly one path from a to x. $R \in T \times T$ is a relation on the set of nodes. To make things simpler, for this question we only consider non-empty trees, that is $T \neq \emptyset$.

This question is about the properties of the relations defining trees. In the table below, make one mark in each row for the corresponding property in the left column, depending on whether all, some, or no relations defining a tree have that property. Put the mark in the corresponding ALL box, if **all relations** defining a tree have the corresponding property, the NONE box, if **no relation** has it, and the SOME box if at least one relation does, and at least one does not.

	ALL	SOME	NONE
reflexive over T			
transitive			
symmetric			
antisymmetric			
asymmetric			

6.3.

Suppose we have a rooted tree (T, R) and a function $\lambda : T \longrightarrow \mathbb{N}$ assigning each node in the tree a natural number as a label.

1. Define the set $T_< \subseteq T$ consisting of all nodes that have a parent and that are labeled with a number greater than that their parent node is labeled with.

   $T_< =$

2. Define a function $\Lambda : T \longrightarrow \mathbb{N}$ that maps every $t \in T$ to the sum of all the labels of nodes below t (i.e. t's children, grandchildren etc.), including t itself.
   (Hint: Here the $\sum_{x \in S} E(x)$ construction might be useful, which sums the $E(x)$ for x bound to all the values in S.)

   $\Lambda : T \longrightarrow \mathbb{N}$

   $t \mapsto$

3. Define the set M of all nodes whose weight is greater than or equal to the sum of the weights of the nodes above it, i.e. their parent, grandparent, etc. all the way back to the root of the tree.

   $M =$

4. Define the set $T_{\mathbb{P}} \subseteq T$ consisting of all nodes that are labeled with a number in $\mathbb{P}$, i.e. a prime number (you do not have to define prime numbers, just use $\mathbb{P}$ in your definition):

   $T_{\mathbb{P}} =$

5. Define the set $A_{x,y} \subseteq T$ consisting of all nodes that have label x and that have at least one child with label y (for $x, y \in \mathbb{N}$):

   $A_{x,y} =$

6. Define the set $B_{x,y} \subseteq T$ consisting of all nodes that have label x and have at least one "grandchild", i.e. one child of a child, with label y (for $x, y \in \mathbb{N}$):

   $B_{x,y} =$

7. Define the set $C_n \subseteq T$, consisting of all nodes that are descendents (children, children of children, children of children of children and so on) of node $n \in T$:

   $C_n =$

6.4.

Suppose we have a rooted tree (T, R) with nodes T, links $R \subseteq T \times T$, and root a as well as a labeling function $\lambda : T \longrightarrow \mathbb{N}$ assigning each node in the tree a natural number.

We want to define a function $L : T \longrightarrow \mathbb{N}$ that computes for each node $n \in T$ the lowest number a node in the subtree rooted at n is labeled with (that subtree includes n itself). If the subtree consists only of n, its label $\lambda(n)$ is the lowest number.

As before, for any non-empty set S of numbers, $\min S$ is the lowest number in that set.

1. Define L using well-founded recursion. (Hint: You may use cases if you like, but it is possible to define this function without an explicit "base case.")

   $$L : n \mapsto$$

2. Define a strict partial order $\prec$ on T such that the poset $(T, \prec)$ is well-founded and your definition of L performs well-founded recursion on that poset. For all $n, n' \in T$...

   $$n' \prec n \iff$$

   No proof is required. It is sufficient that the strict partial order is well-founded and your definition of L conforms to it.

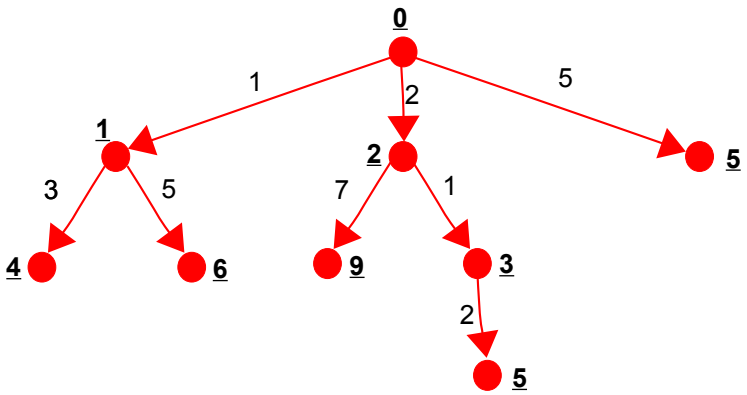
6.5.

Suppose we have a rooted tree (T, R) with nodes T, links $R \subseteq T \times T$.

We are also given a function $w : R \longrightarrow \mathbb{N}$ that assigns each *link* a natural number, let's call it the "weight" of that link.

1. Define a function $W : T \longrightarrow \mathbb{N}$ that maps each node in the tree to its "path weight", that is the sum of the weights of the links on the path from the root to that node.

   Example:



   The numbers next to the links are those assigned to them by the (given) function $w : R \longrightarrow \mathbb{N}$. The underlined numbers next to the nodes are those that your function $W : T \longrightarrow \mathbb{N}$ is supposed to compute in the case of this example.

$W : n \mapsto$

2. Define the set $L \subseteq T$ of *leaf nodes* of a tree, i.e. nodes that do not have any children:

$$L =$$

3. Define the set $M \subseteq L$ of leaf nodes in T with the smallest path weight (you may, indeed should, use L from the previous subtask, and will probably find the $\min$ function useful, where for any set of numbers S, $\min S$ is the minimum number in that set, if it has a minimum):

$$M =$$

6.6.

Suppose we have a directed graph (V, E). We want to define a function $r : V \longrightarrow \mathcal{P}(V)$ that computes for each vertex the set of vertices that can be reached from it by following zero or more directed edges.

We do this using a helper function $r' : V \times \mathcal{P}(V) \longrightarrow \mathcal{P}(V)$ that keeps track of the vertices we have visited already, so that we do not get stuck in cycles. Then r itself simply becomes

$$r(a) = r'(a, \emptyset)$$

The second argument of r' is the set of vertices we have visited already, initially empty.

Your task is to define r' using recursion:

$r' : (a, S) \mapsto$

Hint: Note that the edge relation E can be used to compute all the nodes that can be reached from a given vertex a in one hop: that set is simply the image of a under E, i.e. $E(a)$.

6.7.

Suppose we have a directed graph (V, E). We want to define a function $d : V \times V \longrightarrow \mathbb{N} \cup \{\infty\}$ that computes for each pair of vertices the smallest number of directed edges required to go from one to the other. For any vertex a, $d(a, a) = 0$, and if there is no directed path from a to b, then $d(a, b) = \infty$.

We compute the function d using a helper function $d' : \mathbb{N} \times \mathcal{P}(V) \times \mathcal{P}(V) \times V \longrightarrow \mathbb{N} \cup \{\infty\}$, which keeps track of the vertices we have visited already, so that we do not get stuck in cycles, and also keeps track of the number of steps we have taken from the initial vertex. The second argument is the set of all vertices than can be reached in the next step. With this, the d becomes

$$d(a, b) = d'(0, \{a\}, E(\{a\}), b)$$

The first argument to d' is the number of steps we have taken, the second is the set of vertices we have seen already (and which are therefore at most that many steps away from the start node). The third argument is the set of nodes we can reach from the previous set in one step – note that this is **the *image* of the edge relation** E! Finally, the last parameter is the target vertex.

Your task is to define d' using recursion:

$d' : (n, S, T, v) \mapsto$

The function will need to deal with three cases:
(1) the target vertex v was found after n steps,
(2) the target vertex cannot be reached,
(3) otherwise.

Make sure you identify the conditions for the first two, as well as the return values for all three.

Hint: Note that using the edge relation E can be to compute the next nodes that can be reached at its image, you basically compute all those nodes at once. So if you have a set of nodes A, the nodes that can be reached from A in one step is $E(A)$.

6.8.

Suppose we have a directed graph (V, E), as well as a function $w : E \longrightarrow \mathbb{N}^+$ assigning every edge in the graph a positive natural number, its *edge weight*. The *path weight* of a directed path in the graph is the sum of all edge weights of the edges in that path (if an edge occurs multiple times in a path, its weight is counted for each occurrence).

Using recursion, define a function $R : V \times \mathbb{N} \longrightarrow \mathcal{P}(V)$ such that for any vertex a and any positive integer k, the set $R(a, k)$ is the set of all vertices that can be reached from a by a path with path weight less than or equal to k.

$R : (a, k) \mapsto$

6.9.

Suppose we have a directed graph (V, E), as well as a weight function $w : E \longrightarrow \mathbb{N}^+$ assigning every edge in the graph a positive natural number, its *edge weight*. The *path weight* of a directed path in the graph is the sum of all edge weights of the edges in that path.

We define the *distance* of two vertices in such a graph as the smallest path weight of any directed path connecting them. If there is no such path between two vertices, we say their distance is infinite, ∞. The distance of a vertex to itself is always zero, i.e. $d(a, a) = 0$ for all $a \in V$.

We want to define a function $d : V \times V \longrightarrow \mathbb{N} \cup \{\infty\}$ that computes for each pair of vertices their distance.

We do this using a helper function $d' : V \times V \times \mathcal{P}(V) \longrightarrow \mathbb{N} \cup \{\infty\}$ that keeps track of the vertices we have visited already, so that we do not get stuck in cycles. Then d itself simply becomes

$$d(a, b) = d'(a, b, \emptyset)$$

The third argument of d' is the set of vertices we have visited already, initially empty.

Your task is to define d' using recursion:

$d' : (a, b, S) \mapsto$

For this exercise, you may assume a $\min$ operator that works on $\mathbb{N} \cup \{\infty\}$, such that $\min\{\infty\} = \infty$ and for any $S \subset \mathbb{N}$, it is the case that $S \neq \emptyset \rightarrow \min(S \cup \{\infty\}) = \min S$. Also, you may use addition extended to $\mathbb{N} \cup \{\infty\}$ such that $n + \infty = \infty$ for all $n \in \mathbb{N} \cup \{\infty\}$

However, do keep in mind that $\min \emptyset$ is undefined, so make sure that any set you apply that operator to is not empty.

6.10.

Suppose you have a graph (V, E) with vertices V and edges $E \subseteq V \times V$.

As before, a *path* in this graph is a non-empty finite sequence $v_0 v_1 ... v_n \in V^*$, such that for any $i \in \{0, ..., n-1\}$ we have $(v_i, v_{i+1}) \in E$. The number n, corresponding to the number of edges connecting the vertices in the path (and one less than the number of vertices in the sequence representing it), is called its *length*.

A *cycle* is a path of at least length 1 where the first and the last vertex are the same, so $v_0 = v_n$. A *simple cycle* is a cycle where every vertex occurs at most once, except for the first and last, which occurs exactly twice.

This exercise is about defining a function $C : V \longrightarrow \mathcal{P}(V^*)$ that for any vertex $v \in V$ computes **the set of all simple cycles** starting (and therefore also ending) at v.

We shall do so using a helper function $C' : V \times V^* \times V \longrightarrow \mathcal{P}(V^*)$, such that $C'(v, p, w)$ is the set of all simple cycles that (a) start (and end) at v, (b) then follow the path p, and (c) then continue with vertex w. In other words, $C'(v, p, w)$ is the set of all simple cycles that begin with vpw.

Using this, we can define C as follows (remember that ε represents the empty sequence):

$$C : V \longrightarrow \mathcal{P}(V^*)$$
$$v \mapsto \bigcup_{w \in E(v)} C'(v, \varepsilon, w)$$

Convince yourself that this results in all simple cycles starting at v if C' behaves as described above.

1. Define C' recursively. You may find it useful to look at the *set* of all vertices occurring in a path $p \in V^*$. You can use the notation $set(p)$ for this purpose, i.e. if p is the path $v_0 v_1 ... v_n$, then $set(p)$ is the set $\{v_0, v_1, ..., v_n\}$.

   $C' : V \times V^* \times V \longrightarrow \mathcal{P}(V^*)$

   $v, p, w \mapsto$

2. In order to ensure that C' terminates, we require a **well-founded strict order** $\prec$ of its arguments, such that for any (v, p, w) that C' is called on, it will only ever call itself on $(v', p', w') \prec (v, p, w)$. Define such an order:

   $(v', p', w') \prec (v, p, w) \iff$

   Hint: A correct answer to this question must have three properties.
   1. It must be a strict order.
   2. It must be well-founded, i.e. there cannot be an infinite descending chain in that order.
   3. Your definition of C' must conform to it, i.e. any recursive call in it must be called on a smaller (according to the order) triple of arguments.

6.11.

Suppose we have a graph (V, E), as usual with vertices V and edges $E \subseteq V \times V$, as well as a function $w : E \longrightarrow \mathbb{N}$ assigning each edge a natural number as a *weight*.

1. Define the set $E_{\leq k} \subseteq E$ consisting of all edges in E with weight not more than k:

   $E_{\leq k} =$

2. Define the function $R : V \times \mathbb{N} \longrightarrow \mathcal{P}(V)$, such that $R(v, n)$ is the set of all vertices in$V$ that can be reached from v in exactly n steps, and $R(v, 0) = \{v\}$.

   $R : V \times \mathbb{N} \longrightarrow \mathcal{P}(V)$

   $v, n \mapsto$

3. Define the relation $P \subseteq V \times V$ such that for any two vertices $v, w \in V$, it is the case that $(v, w) \in P$ iff there is a path from v to w in the graph (V, E).

   $P =$

4. Define the relation $D \subseteq V \times V$ on the vertices in V such that for any two vertices $v, w \in V$ it is the case that $(v, w) \in D$ iff there is a path p from v to w and another path q from w to v that has the same length as p.

   $D =$

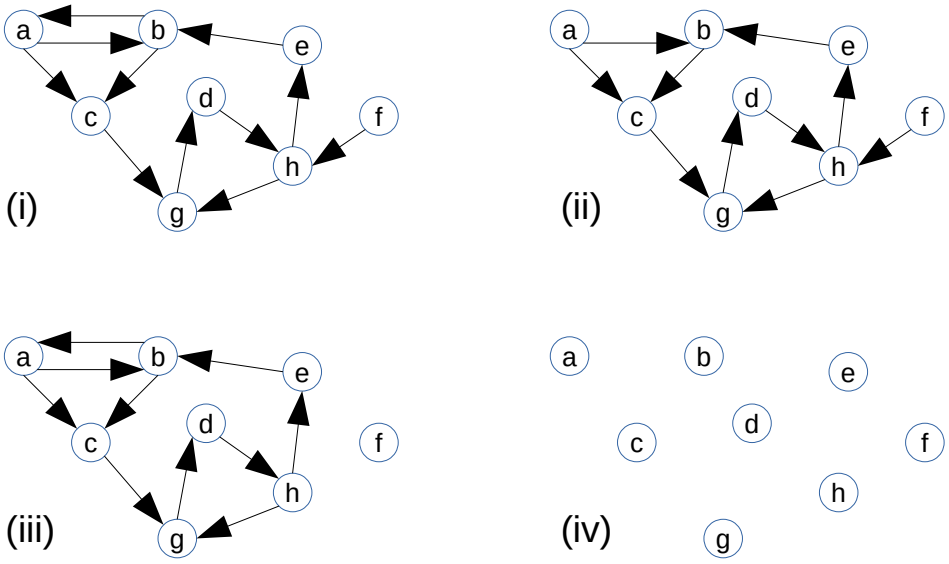


Figure 3: Four graphs over the same set of vertices, and different longest path lengths.

6.12.

Suppose we have a directed graph (V, E) with vertices V and edges $E \subseteq V \times V$. In this task, you will be asked to compute the length of the *longest* simple path in such a graph.

First, a few definitions: A *path* is a non-empty, finite sequence $v_0 v_1 v_2 ... v_n$ of vertices $v_i \in V$ such that between any two successive v_i and v_{i+1} there is an edge in E from v_i to v_{i+1}, or more formally, for all $i = 0...n-1, (v_i, v_{i+1}) \in E$. The *length* of this path would be n (the number of edges in the path, or one less than the number of vertices in it). An empty path, of length 0, would be one consisting of only one vertex. A *simple path* is a path in which every vertex occurs at most once, in other words a simple path is one that is not a cycle and also does not contain a cycle.

Given a graph (V, E), we are interested in the length of the longest simple path (we shall just say "longest path" from now on) in it. There may be more than one longest path.

As an example, consider the graphs in Fig. 3. In graph (i), the longest path has length 7, and it is **fhebacgd**.[2] In graph (ii), which is like the one before except the link from **b** to **a** is missing, the longest path is now of length 6, and there are several: **abcgdhe**, **acgdheb**, **fhebcgd**. Graph (iii) is also like (i), except now we have removed the link from f to h, isolating vertex f, and again the longest path is of length 6 – and again, there are several longest paths. Finally, in example graph (iv), all edges have been removed, which means the longest path has length 0, and there are eight of them.

2 Since the path has length $\#V - 1$, it is also a Hamiltonian path.

Define $LP \in \mathbb{N}$, the length of the longest path given a graph (V, E), in two steps.

Note: You might find the function $\max$ useful, which computes the maximum number in a finite, **non-empty** set of natural numbers. It is undefined for the empty set or for infinite sets. So $\max\{2, 3, 11\} = 11$, and $\max\{2k : k \in \mathbb{N}\}$ as well as $\max \emptyset$ are undefined.

1. We start with a function $L : V \longrightarrow \mathbb{N}$ that computes, for each vertex, the length of the longest path *starting at that vertex.* It is defined using a helper function
   $L' : V \times \mathcal{P}(V) \longrightarrow \mathbb{N}$
   that uses its second argument to keep track of the vertices it already visited. (Note that the "current" vertex is already in that set.)

   $L : V \longrightarrow \mathbb{N}$
   $v \mapsto L'(v, \{v\})$

   Define L'.

   $L' : V \times \mathcal{P}(V) \longrightarrow \mathbb{N}$

   $v, S \mapsto$

2. Using the function $L : V \longrightarrow \mathbb{N}$, define the value $LP \in \mathbb{N}$ which is the length of the longest simple path in the graph (V, E):

   $LP =$

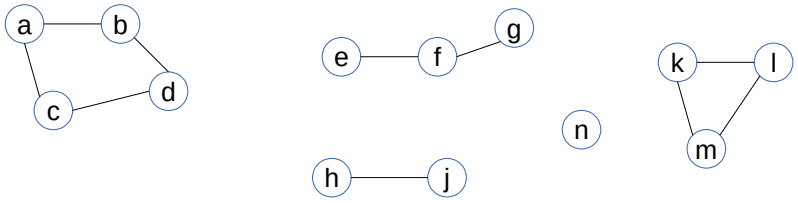


Figure 4: Example, villages and roads between them.

6.13.

Suppose we have a graph (V, R) with vertices V representing villages and edges $R \subseteq V \times V$ representing the roads between them, similar to the one in Fig. 4 above. The road relation is *symmetric*, that is $\forall v_1, v_2 \in V \ ((v_1, v_2) \in R \Leftrightarrow (v_2, v_1) \in R)$.

1. In the example above in Figure 1, we have the vertices
   $V = \{a, b, c, d, e, f, g, h, j, k, l, m, n\}$.
   Give the edge relation R for the example:

   $R =$

2. Define, for any such graph (V, R) (not just for the example above!), the set D of dead-ends, i.e. the set of all villages one cannot leave (i.e. go to different village) at all by road, or by at most one road. In the example, this set would be $D = \{e, g, h, j, n\}$.

   $D =$

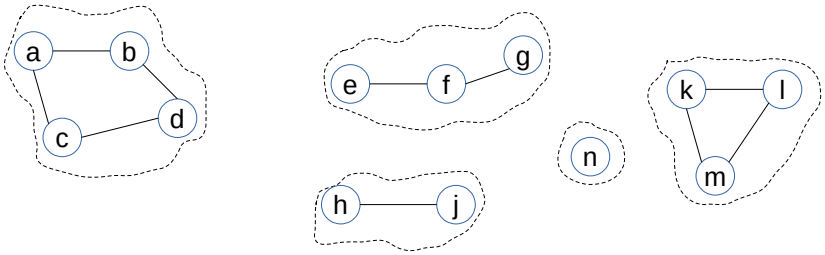


Figure 5: Grouping villages into islands

6.14.

Suppose the villages and roads in (V, R) from exercise 6.13 are located on a set of islands, and each island corresponds exactly to the villages that can be reached from one another by road (i.e. all the villages on an island are connected by roads, not necessarily directly), as shown for our example in Fig. 5.

Define for any such (V, R) the set A of islands, that is a set of sets of villages that can reach each other by road. In the example, we would have

$$A = \{\{a, b, c, d\}, \{e, f, g\}, \{h, j\}, \{k, l, m\}, \{n\}\}.$$

$A =$

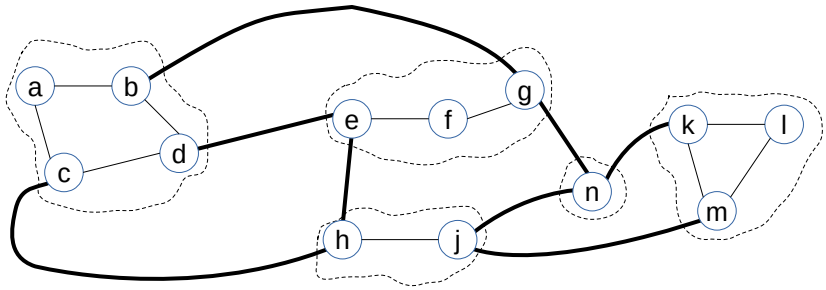


Figure 6: Our example villages and roads on the archilepago, with ferry connections between islands drawn in bolder lines.

6.15.

Suppose that in addition to the villages in V and the symmetric relation of the roads connecting them R from exercise 6.13 we also have another *symmetric* relation $F \subseteq V \times V$ of ferry connections connecting villages on different islands, as in Fig. 6. We assume that $F \cap R = \emptyset$, that is that there is no ferry connecting villages that are also connected by a road.

Define the set $Ports$ of ports, that is villages that have at least one ferry connection. In the example, this would be $Ports = \{b, c, d, e, g, h, j, k, m, n\}$.

$Ports =$

6.16.

Suppose the villages V, the roads $R \subseteq V \times V$, and the ferry connections $F \subseteq V \times V$, as in exercise 6.15. The combined connections between all the villages, using roads and ferries, is the relation $E = R \cup F$. (Recall that R and F are disjoint, so $R \cap F = \emptyset$.)

We want to define a function $P : V \times V \longrightarrow \mathcal{P}(V^*)$ such that $P(v_1, v_2)$ returns all cycle-free paths from v_1 to v_2 in E, that is all paths such that every vertex (village) occurs at most once in it. If $v_1 = v_2$, it returns a set containing only the empty path from v_1 to itself, that is $P(v_1, v_1) = \{v_1\}$.

We define P using a helper function $P' : V^* \times V \times V \longrightarrow \mathcal{P}(V^*)$ that builds the path as it searches for its destination:

$$P : V \times V \longrightarrow \mathcal{P}(V^*)$$

$$v_1, v_2 \mapsto P'(\varepsilon, v_1, v_2)$$

Here, the first argument to P' is the path traveled so far (important for producing the output, and also as a record of the vertices you have been to already), the second is the next vertex to visit, and the third is the destination.

1. Define P'. You might find it useful to talk about the set of all vertices in a sequence of vertices (that is, in a path) – if $p \in V^*$, then you can use $\mathrm{set}(p)$ to describe the set of all vertices that occur in p.

   $$P' : V^* \times V \times V \longrightarrow \mathcal{P}(V^*)$$

   $$p, v, w \mapsto$$

2. Define the set $\Pi \subseteq V^*$ of all non-cyclic paths in (V, E), as computed by P:

   $$\Pi =$$

6.17.

Suppose we have a function $d : E \longrightarrow \mathbb{R}^+ \setminus \{0\}$ from the combined connections to the positive real numbers, signifying for each $(v_1, v_2) \in E$ the time it takes to travel from v_1 to v_2 (by either road or ferry, depending on whether $(v_1, v_2) \in R$ or $(v_1, v_2) \in F$). For any $(v_1, v_2), (v_2, v_1) \in E$, it is NOT guaranteed that $d(v_1, v_2) = d(v_2, v_1)$.

Given a path $v_0 v_1 v_2 ... v_n$ of length n in E, we want to measure it regarding the total time it takes and also the number of ferry connections in the path. The function $M : \Pi \longrightarrow \mathbb{N} \times \mathbb{R}^+$ computes for each non-cyclic path $p \in \Pi$ (the set of all non-cyclic paths from exercise 6.16) a pair (n, r), where n is the number of ferry connections and r the sum of the travel time of all the connections, by road or ferry, in the path.

For every $v \in V$, applying M to the empty path v results in $(0, 0)$, i.e. $\forall v \in V \ (M(v) = (0, 0))$.

Define M.

$M : \Pi \longrightarrow \mathbb{N} \times \mathbb{R}^+$

$p \mapsto$

It may be useful to add pairs of numbers. You may use a lifting of addition to tuples, i.e. a $+$ operation on tuples that is defined like this: $(a, b) + (c, d) = (a + c, b + d)$

Hint: you might find it useful to distinguish cases where p has the form v (i.e. it is an empty path), from cases where it has the form $v_1 v_2 q$, i.e. a path with at least two vertices (i.e. of length at least 1).

7. Logic

7.1.

Find DNFs for the following formulae. Write “none” if a formula has no DNF.

1. $\neg((r \vee q) \leftrightarrow (q \vee p))$

2. $\neg((p \to q) \vee (q \to r) \vee (r \to p))$

3. $(p \to q) \wedge (q \to r) \wedge (r \to p)$

4. $(r \vee q) \to (q \vee p)$

5. $(p \to q) \to ((q \to r) \vee (r \to p))$

6. $(p \to (q \wedge r)) \vee (q \to (p \wedge r)) \vee (r \to (p \wedge q))$

7. $(r \vee q) \rightarrow (q \wedge p)$

8. $(p \rightarrow q) \rightarrow ((q \rightarrow r) \wedge (r \rightarrow p))$

7.2.

Let $E = p \to (q \to r)$ be a Boolean formula with three elementary letters, p, q, r.

1. Find an equivalent DNF for E.

2. Find an equivalent *full* DNF for E. As a reminder, a DNF is full iff every letter (in this case, p, q, r, occurs in each of the basic conjunctions.

3. Find an equivalent form of E using only the $\overline{\wedge}$ connective. Remember, that it was defined as $\alpha \mathbin{\overline{\wedge}} \beta = \neg(\alpha \wedge \beta)$, and that $\neg\alpha = \alpha \mathbin{\overline{\wedge}} \alpha$, so it's probably a good idea to first find a form without disjunctions (or), and then go from there. Careful: The $\overline{\wedge}$ connective is NOT associative, i.e. it is NOT the case that $(\alpha \mathbin{\overline{\wedge}} \beta) \mathbin{\overline{\wedge}} \gamma$ is equivalent to $\alpha \mathbin{\overline{\wedge}} (\beta \mathbin{\overline{\wedge}} \gamma)$!

4. Fill in the truth table for E.

p	q	r	E
1	1	1	
1	1	0	
1	0	1	
1	0	0	
0	1	1	
0	1	0	
0	0	1	
0	0	0	

7.3.

Show that $\overline{\wedge}$ is not associative.

7.4.

As we saw in the lecture on quantificational logic, $\exists x \forall y Rxy$ always implies $\forall y \exists x Rxy$ for any relation *R*. The converse, however, is not necessarily the case: $\forall x \exists y Rxy$ does not always mean that $\exists y \forall x Rxy$ is true.

Define a binary relation R over a non-empty universe D (that you also need to define) such that $\forall x \exists y Rxy$ is true, and $\exists y \forall x Rxy$ is false.

Hint: Keep in mind that the $\forall$ and $\exists$ operators are quantified over D.

D = _____

R = _____

7.5.

True or false?

1.	$\forall x \in \mathbb{R}\ (x^2 > 0)$	true	false
2.	$\forall x \in \mathbb{R}\ (\exists n \in \mathbb{N}\ (x^n \geq 0))$	true	false
3.	$\exists a \in \mathbb{R}\ (\forall x \in \mathbb{R}\ (ax = x))$	true	false
4.	$\forall X \in \mathcal{P}(\mathbb{N})\ (X \subseteq \mathbb{R})$	true	false
5.	$\forall n \in \mathbb{N}\ (\exists X \in \mathcal{P}(\mathbb{N})\ (\#X \leq n))$	true	false
6.	$\exists X \in \mathcal{P}(\mathbb{N})\ (\forall n \in \mathbb{N}\ (\#X \leq n))$	true	false
7.	$\forall X \in \mathcal{P}(\mathbb{N})\ (\exists n \in \mathbb{Z}\ (\#X = n))$	true	false
8.	$\forall n \in \mathbb{Z}\ (\exists X \in \mathcal{P}(\mathbb{N})\ (\#X = n))$	true	false
9.	$\forall n \in \mathbb{N}\ (\exists X \in \mathcal{P}(\mathbb{N})\ (\#X = n))$	true	false
10.	$\forall m \in \mathbb{Z}\ (\exists n \in \mathbb{Z}\ (m = n + 5))$	true	false
11.	$\exists m \in \mathbb{Z}\ (\forall n \in \mathbb{Z}\ (m = n + 5))$	true	false
12.	$\exists n \in \{k \in \mathbb{N} : k > 2\}\ (\exists a, b, c \in \mathbb{N}^+\ (a^n + b^n = c^n))$	true	false [3]
13.	$\forall x, y \in \mathbb{R}\ (x < y \to \exists m \in \mathbb{R}\ (x < m < y))$	true	false
14.	$\exists a, b, c \in \{0, 1\}\ (a \barwedge (b \barwedge c) = (a \barwedge b) \barwedge c)$	true	false
15.	$\forall a, b, c \in \{0, 1\}\ (a \barwedge (b \barwedge c) = (a \barwedge b) \barwedge c)$	true	false
16.	$\exists a \in \mathbb{R}\ (\forall x, y \in \mathbb{R}\ (ax = y))$	true	false
17.	$\forall x, y \in \mathbb{R}\ (\exists a \in \mathbb{R}\ (ax = y))$	true	false

3 I would not ask this one in an exam, but perhaps you recognize it. If not, look up “Fermat’s last theorem”.

8. Proofs

8.1.

Show that a relation $R \subseteq A \times A$ is transitive iff $\forall (a, b) \in R \ (R(b) \subseteq R(a))$

Hints:

- It makes sense to split the proof into two parts: (1) showing that transitivity of R implies that for every $(a, b) \in R$ we have $R(b) \subseteq R(a)$ and then (2) showing that $R(b) \subseteq R(a)$ for every $(a, b) \in R$ implies that R is transitive.
- Note that $A \subseteq B$ iff $\forall x(x \in A \rightarrow x \in B)$.

8.2.

Given two transitive relations R and S, is the relation $R \cap S$ transitive?

If yes, prove it. If no, provide a counterexample.

8.3.

Given two transitive relations R and S, is the relation $R \cup S$ transitive?

If yes, prove it. If no, provide a counterexample.

8.4.

Yet another property that is equivalent to transitivity:

A relation R is transitive iff $R \circ R \subseteq R$.

1. Implement transitive? in Lab 1 using this property.
2. Prove it. Once again, it might be best to prove the two directions separately.

These we discussed in the lecture:

8.5.

Show that for any integer x, if x is even, then x^2 is even.

8.6.

For any two integers a, b, we say that $a \text{ divides } b$, and write $a|b$, iff there is an integer k such that $ak = b$.

Show that for any three integers a, b, c, if $a|b$ and $b|c$, then $a|c$.

8.7.

Show that for any two injections $f : A \hookrightarrow B$ and $g : B \hookrightarrow C$, their composition $g \circ f : A \longrightarrow C$ is injective.

8.8.

Show that every multiple of 4 equals $1 + (-1)^n(2n - 1)$ for some $n \in \mathbb{N}$.

Hint: If k is a multiple of 4, it means there is an integer $a \in \mathbb{Z}$ such that $k = 4a$. For this proof, it helps to use the cases $a = 0$, $a > 0$, and $a < 0$.

8.9.

Show that for any integers $x, y \in \mathbb{Z}$, if 5 does not divide xy then 5 does not divide x, and it also does not divide y.

Hint 1: The logical negation of (A and B) is (not A **or** not B).

Hint 2: This proof is best done using cases.

8.10.

Show that the number $\sqrt{2}$ is irrational.

Hint 1: The opposite of a number being irrational is that it can be represented as a fraction $\frac{a}{b}$ of integers a, b. It is useful to require that the fraction be fully reduced (that's how we will produce the contradiction in this case), i.e. the two integers do not have a common divisor.

Hint 2: In particular, they cannot both be even, because that would mean that 2 is a common divisor.

Change history

Date	Modifications
2021.03.24	Remove use of $\wedge$ in 1.3 and explain $a \mid b$ in 1.4.
2021.03.27	Added defs for $a \mid b$, $a \perp b$, and mod in 2.2, 2.3, 2.4, 2.5, and 2.8.
2021.05.18	fixed inconsistencies in 6.17
2021.05.19	fixed error in 7.4