- Connectives:

$\neg P$	Negation
$P \wedge Q$	Conjunction
$P \vee Q$	Disjunction
$P \to Q$	Implication
$Q \to P$	Converse
$\neg Q \to \neg P$	Contrapositive
$\neg P \to \neg Q$	Inverse
$P \leftrightarrow Q$	Biconditional

-Tautologies & Contradictions:
$P \to P$ and $P \vee \neg P$ are *always* true
$P \wedge \neg P$ is *always* false.

-Logical Equivalences:
<use other sheet>

-Predicate Logic:
Rather than $x>3$ we can say $P(x)$:

$P(2)$ = False
$P(42)$ = True

Predicate values give a T/F response.

-Quantifiers:
$\forall x$: for all x
$\exists x$: there exists x
$x \in U$: in the universal domain

$\forall x.(P(x) \wedge P(y)) = \forall x.(P(x) \wedge \forall x.(P(y))$
$\exists x.(P(x) \vee P(y)) = \exists x.(P(x) \vee \exists x.(P(y))$
(does *not* work other way round!)

-Sets:
Look to logical equivalence sheets, however,

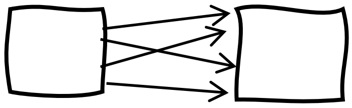
U = True,
∅ = False,
∪ = V,
∩ = ^,
A' = ¬.

A' or $\overline{A}$ is the complement of set A.
$U = \{1,2,3,4,5\}$, $A = \{1,2\}$, then $A' = \{3,4,5\}$
$|A|$ = cardinality of A, how many elements in
$P(A)$ = powerset, all possible subsets.

-Proving Set Equalities:
Four ways:

- Venn Diagram,
- membership table,
- containment proof *(make RHS equal to LHS then other way, both contained in each other and therefore equal),*
- set comprehension and logical equivalences

-Functions:
f : [domain] -> [codomain]



domain maps to *one* value in the codomain, otherwise not a function. Codomain can have 'free' values tho.
$(f1 + f2)(x) = f1(x) + f2(x)$ (same for *)

-Injective, bijective, surjective and inverse:
Strictly increasing: $x1<x2$ then $f(x1)< f(x2)$

In: one-to-one. Each element in x maps to a unique element of y.
Sur: each element of the codomain has *at least* one preimage (value in domain)
Bi: both of the above.

Inverse functions: the function *must* be bijective for it to work. Steps:

replace 'f(x)' w 'y'
solve for x
change 'x' to 'f'(x)' and 'y' to 'x'

-divisibility
$a|b$ – "a is a factor of b" and "b is a *ble of a"

$a = b * \text{quotient} + \text{remainder}$

-Primes:
Fundamental theorem of arithmetic: every positive integer can be written as a unique product of primes. Eg: $42 = 2^1 * 3^1 * 7^1$.

GCD and Prime factorisation: can do as above. Also using the Euclidian Algorithm:

$78 = 66(1) + 12$
$66 = 12(5) + 6$
$12 = 6(2) + 0$
$\therefore GCD(78) = 6$

Congruence: $a \equiv b \pmod{c}$
Iff a mod m= b mod m.
$6 \equiv 11 \pmod{5}$: 6-11 = -5 and -5 is divisible by 5

-Matrices:
Can sum (+) two matrices is they are the same size

A	B	+	E	F	=	A+E	B+F
C	D		G	H		C+G	D+H

-Matrices (cont.)

Multiplication:

A	B	*	E	F	=	AE + BG	AF + BH
C	D		G	H		CE + DG	CF + DH

Transpose:

A = [i,j], A^T = [j, i]

A	B	C
D	E	F

A	D
B	E
C	F

-Proofs:

Direct: P -> Q

Indirect: ¬Q -> ¬P (assume Q is False)

Trivial: P -> Q, when Q = True

Contradiction: assume negation cannot hold

By Cases: like in AE

Vacuous: If P doesn't hold then done

Existence: ∃x False, or ∀x True

Induction: find a base case that holds (base case) and then use that to prove that all other steps hold (inductive step).

Recursive functions: select a base case and work your way down to it.

Recursively defined structures: Trees!

-Counting:

Product rule: |A| = m & |B| = n, then there are mn ways to combine one element from A with one from B. Works for 2+ products.

Sum rule: |A| = m & |B| = n, and A & B are disjointed then there is n + m ways to choose an element from A or B.

Inclusion-exclusion principle: The sum rule but when A and B have overlapping elements. |A∪B| = |A| + |B| − |A∩B|.

The pigeonhole principle: if n objects are placed in k containers, then at least one container has ceil(n/k) objects (round up to nearest int).

-Permutations:

An ordered arrangement of objects.

P(n,r) is an ordered arrangement of size r of a set of size n.

$$P(n,r) = n!/(n-r)!$$

-Combinations:

An *un*ordered arrangement of objects.

$$C(n,r) = n!/(r!\cdot(n-r)!)$$

-Combinations with repetitions:

B = number of objects to be selected

C = number of 'containers' that can be chosen from (minus one)

A = B + C

$$Combinations = A! / (B! * C!)$$

Or

$$C(n+r-1, r).$$

-Permutations with indistinguishable objects:

n1 are indistinguishable

n2 are indistinguishable

$$n!/(n1!\cdot n2!)$$

-Probability:

Theorising propositions on the likelihood of outcomes.

Eg. Drawing a 4 from a deck of cards: 4/52 or 1/13

$0 \leq P[A] \leq 1$ for all events $A \subseteq \Omega$;

$P[\Omega]=1$;

if A and B are mutually exclusive, then $P[A \cup B]=P[A]+P[B]$;

-Conditional probability:

If we are given the base B, then the probability of A is:

$$P[A/B] = P[A \cap B] / P[B]$$

A and B are independent iff P[A ∩ B] = P[A] * P[B]. If this is the case then P[A|B] = P[A].

Bayes' rule:

$$P[A/B] = \frac{P[B/A] * P[A]}{P[B]}$$

Random variables:

*E[X] = [sum of, x in domain X] P[X=x] * x*

-Graphs:

Undirected graphs:

G = (V,E) – *finite vertices, edges in vertices, of size 2.*

Cannot self-loop, cannot have multiple edges.

Adjacent: when there is an edge connecting

Degree: how many edges a vertex has.

Directed Graphs:
Edges got directions! An ordered pair (x,y)
In and Out degree, degree but measuring how many going in (+) and how many going out (-):

$$Deg^{+}(a) = 2.$$

Isomorphic graphs:
Graphs that have same structure but a different arrangement (look different).
To check you should see if a bijection can be made between the graphs. Look at the degree of each vertex and see if is it correlates.

Connectivity:
How many 'steps' (or vertices) do you need to cross to get from A to B.
A graph is *connected* if every pair in the graph can be joined by a path.
A *tree* is when it is connected and acyclic (no cycles)

-Relations:
Between elements in a set or between different sets. (e.g. Students to subjects)
A relation between a and b is written as aRb.
Binary: subset of cartesian product.
N-ary: a set of n-tuples

N = 1 : urinary
N = 2 : binary
N = 3 : ternary
N = … : n-ary

Can be represented as a directed graph.

Properties:

- Reflexive: when every element is related to itself
- Symmetric: if aRb then bRa
- Transitive: if aRb and bRc then aRc

Combining relations:
R over A×B, S over B×C
R combined with S *(S∘R)* over A×C such that (a,c)∈S∘R if and only if there exists b∈B such that (a,b)∈R & (b,c)∈S

Closures:
The closure of relation R with respect to some property P is given by the relation S where S is R union the minimum number of tuples that ensures property P holds. Can be Symmetric, Transitive or Reflexive.

-Partial orders:
R over SxS is a partial order if it is reflexive, anti-symmetric and transitive.
Denoted as: (S, ⊑).

-Lexicographical ordering:
Effectively a generalisation of alphabetical ordering. All "A….." comes before all "B……"