Course material overlap with nearby USyd units

This summary describes overlap in course material. It locates the unit MATH2080/2980 within the surrounding algebra curriculum. Nearby courses in algebra, number theory, and logic may shift emphasis as offerings are revised. Overlap is not duplication: the same object may be introduced as an example in one unit, used as a computational tool in another, and studied structurally in a third.

This discussion is a conceptual map, not a substitute for formal enrolment advice or the current handbook. In particular, prerequisites, exclusions, advanced-unit arrangements, and even detailed syllabi should be checked for the year in which a student plans to enrol. The comparisons below describe the role played by shared material and indicate what new viewpoint each unit adds.

MATH2022/2922 and MATH2080/2980

MATH2022/2922, Linear and Abstract Algebra, develops vector spaces, matrices, linear transformations, kernels, ranges, isomorphism, eigenvalues and eigenvectors, and uses matrix and permutation groups to motivate abstract group theory. MATH2922 also puts more emphasis on rigour and on advanced linear-algebra structure such as inner products, quadratic forms, and normal forms.

The characteristic question in linear algebra is how a linear transformation can be understood after choosing suitable bases. Matrices encode the maps, while kernels, images, eigenvalues, and normal forms reveal structure that is independent of the coordinates. This is excellent preparation for abstract algebra: it trains us to distinguish an object from a particular presentation of it and to regard isomorphic objects as structurally the same.

The main overlap with MATH2080/2980 is in examples and first group-theoretic language:

- matrix groups, especially $\mathrm{GL}_n(F)$ and subgroups defined by equations such as determinant conditions;
- permutation groups, cycle notation, transpositions, signs, and symmetry examples;
- the use of kernels, images, and isomorphism as structural tools;
- fields and vector spaces as familiar settings in which algebraic structures live.

MATH2080/2980 takes these examples as a starting point and develops group theory more systematically: cyclic, dihedral, and quaternion groups; subgroups; homomorphisms; actions; cosets; quotients; isomorphism theorems; composition series; Sylow theory; and finite abelian groups. It also adds a substantial computational strand, using concrete examples and computer algebra to test and explore structure.

Thus a student may encounter permutations or matrix groups in both units but be asked different questions. In MATH2022/2922 a group often organizes linear examples or symmetries; here it becomes the primary object. We ask how its subgroups fit together, how it acts on sets, which quotients and composition factors it has, and which invariants determine it up to isomorphism. Familiarity with matrices and permutations is helpful, but the purpose of revisiting them is to move from concrete calculation to general group-theoretic arguments.

MATH2080/2980 and MATH2088/2988

MATH2088/2988, Number Theory and Cryptography, develops elementary number theory as the mathematics behind public-key cryptography. Its official description highlights the Euclidean algorithm, Fermat's little theorem, the Chinese remainder theorem, Mobius inversion, RSA, Elgamal, Diffie–Hellman, computational complexity, primes, prime factorisation, modular arithmetic, divisors and multiplicative functions, powers, and discrete logarithms. MATH2988 is the advanced version of MATH2088, with the same lectures and more advanced tutorial and computer-laboratory material.

Here the motivating questions are arithmetic and algorithmic. One wants not only to know that an inverse or a prime factor exists, but to compute it, to estimate the cost of doing so, and to understand why the difference between an easy forward operation and a difficult inverse operation can support a cryptosystem. Congruences are therefore studied both as mathematical relations and as the data type on which the algorithms operate.

The main overlap with MATH2080/2980 is in arithmetic structure and exact computation:

- divisibility, greatest common divisors, Bezout identities, and Euclidean algorithms;
- modular arithmetic, congruence classes, units modulo n, and the Chinese remainder theorem;
- powers, orders of elements, cyclic groups, and discrete logarithm language;
- Magma-based exploration of number-theoretic and algebraic algorithms.

MATH2088/2988 keeps the emphasis on arithmetic algorithms and cryptographic applications: how one computes with congruences, powers, primes, factorisation, and cryptosystems, and why some computations are feasible while others are hard. MATH2080/2980 uses many of the same arithmetic tools but places them inside a broader structural framework: groups, rings, ideals, quotient objects, polynomial quotient rings, finite fields, Groebner bases, and symmetry.

For example, the group of units $(\mathbb{Z}/n\mathbb{Z})^\times$ appears naturally in both settings. Number theory asks how to exponentiate in it, determine orders, and use those operations in RSA or Diffie–Hellman. Abstract algebra asks how its group structure depends on n and how it exemplifies general statements about finite abelian groups. The overlap therefore reinforces essential computations, while the surrounding goals remain different.

MATH2080/2980 and MATH3066

MATH3066, Algebra and Logic, combines algebra with foundations. Its official description emphasizes field and ring axioms, field extensions, impossibility arguments for classical construction problems, quotient rings, a construction of the real numbers from Cauchy sequences of rationals modulo null sequences, first-order logic and set theory, propositional and predicate calculi, consistency and completeness, computability, Turing machines, the halting problem, and undecidability.

The unifying theme is the relation between formal language and mathematical possibility. Algebraic axioms let us prove that certain constructions are possible or impossible; logical systems let us ask which statements are provable; models and algorithms let us ask which questions are decidable. A quotient construction is important here not merely as a calculation but as a precise way to identify objects that should count as equivalent.

The main overlap with MATH2080/2980 is in algebra as an axiomatic language:

- groups, rings, integral domains, and fields as structures defined by axioms;
- quotient constructions, especially quotient rings and ideals;
- field extensions and the algebraic language behind construction and impossibility arguments;
- logical fluency: definitions, quantified statements, deduction, and proof;
- computability and algorithmic limits as a conceptual counterpart to exact computation.

MATH3066 uses algebra partly to ask foundational questions: what follows from axioms, what can be constructed, what can be proved, and what can be decided by an algorithm. MATH2080/2980 overlaps in its attention to axioms, quotient structures, and computation, but it keeps the main emphasis on classifying and computing with algebraic objects themselves.

Accordingly, proof skills transfer strongly between the units even where the topics diverge. Quantifier order matters in definitions such as normality, irreducibility, and computability; a counterexample must address exactly the statement being denied; and quotient objects require a well-definedness argument. MATH2080/2980 develops these habits through algebraic structures, whereas MATH3066 carries them toward formal logic and limitations of computation.

MATH2080/2980 and MATH3962/4062

MATH3962/4062, Rings, Fields and Galois Theory, studies the algebra behind polynomial equations: rings, fields, field extensions, polynomial quotient rings, with the final application found in Galois theory. Its official description frames rings as the basic theoretical tool generalizing fields, then applies ring and quotient-ring ideas to polynomial rings before moving toward field extensions and Galois theory.

The central change of viewpoint is from manipulating one polynomial to studying all of its roots and the symmetries among them. Rings organize divisibility and factorization; quotient rings adjoin roots; field extensions measure the algebra needed to contain those roots; and automorphism groups record the relations that remain invisible in the coefficients alone.

The main overlap with MATH2080/2980 is in the ring-theoretic bridge:

- rings, subrings, units, zero divisors, integral domains, and fields;
- ring homomorphisms, ideals, kernels, and quotient rings;
- arithmetic in polynomial quotient rings such as $F[x]/(f)$;
- irreducibility, factorization, and the Euclidean algorithm in $F[x]$.

MATH2080/2980 introduces these topics after the group-theory core and emphasizes computation with finite rings, ideals, quotient representatives, polynomial arithmetic in several variables, and Groebner-basis methods. MATH3962/4062 continues in a more theoretical direction: field extensions, splitting fields, and Galois theory.

The handoff is especially visible in quotients $F[x]/(f)$. In this course they provide finite fields and a concrete setting for arithmetic: reduce powers, compute inverses by Bezout, and decide how f factors. In a later fields and Galois course, the same construction is interpreted as adjoining a root, and one studies extension degree, splitting fields, and automorphisms. Likewise, the group theory developed here supplies the language for Galois groups and their subgroup structure.

How to read the sequence

A potentially useful way to visualize the mathematical progression is:

$$\text{MATH2022/2922} \to \text{MATH2080/2980} \to \text{MATH3962/4062}.$$

The first arrow moves from linear algebra and motivating group examples to a full course in groups, rings, and computational algebra. The second arrow moves from introductory ring and polynomial quotient methods to the theory of fields and polynomial equations. Thus MATH2080/2980 is the hinge: it turns the abstract structures first glimpsed in linear algebra into objects of study in their own right, then prepares the ring and field language needed for Galois theory.

MATH2088/2988 sits beside this sequence rather than simply before or after it. It overlaps most strongly with the bits of number theory and algorithmic parts of MATH2080/2980, but it develops those tools toward cryptography and computational number theory rather than toward general group, ring, and quotient structure.

MATH3066 is another neighboring course rather than a direct continuation. Its algebra overlap is strongest around axioms, rings, fields, quotient rings, field extensions, and proof, but its destination is logic, foundations, computability, and undecidability.

Another way to read the map is by following a single recurring construction. A kernel first appears in linear algebra as the vectors sent to zero. In group and ring theory it becomes the normal subgroup or ideal that determines a quotient and drives an isomorphism theorem.

In number theory, the same quotient language underlies congruence arithmetic and CRT. In field theory, kernels and quotients construct extensions, while in logic an equivalence relation and its quotient can be part of the construction of a new mathematical universe. The common vocabulary is intentional; each course asks it to do a different job.

The arrows should therefore be read as a description of mathematical emphasis, not as a claim that every student must follow one linear route. The current handbook and degree rules determine actual prerequisites and permitted combinations. Within those rules, students can use the distinctions above to decide whether their next interest is computational number theory and cryptography, structural ring and field theory, or algebra in conversation with logic and foundations.