MA4N1 Project Outline - Artin-Wedderburn

Ross Truscott, Joseph Bettridge, Jiejhui Jiang

November 6, 2025

The Result

We aim to prove the Artin-Wedderburn theorem, the structure theorem for semi-simple Artinian rings as stated below (note that for us rings are unital but not necessarily commutative).

Theorem 1 (Artin-Wedderburn)**.** Let R be a left Artinian semi-simple ring. Then there exist division rings $D_1, \ldots, D_m$ and positive integers $n_1, \ldots, n_m$ such that,

$$R \cong \bigoplus_{i=1}^{m} M_{n_i}(D_i).$$

Moreover, R has exactly m non-isomorphic simple submodules S_i and $D_i \cong \mathrm{End}_R(S_i)$ for each $i \in \{1, \ldots, m\}$.

This result is already in mathlib, however, we intend to approach it in a more "mathematical" fashion, along the lines of the following proof based on the one I saw in my UG, but it is similar to the one in [1].

The Proof

Proving the main result will involve proving various lemmata, which we present below.

Lemma 1 (Schur's Lemma)**.** If M is a simple R-module, then $\mathrm{End}_R(M)$ is a division ring.

Proof. Let $\varphi \in \mathrm{End}$. Then $\ker \varphi \lhd M$ and so φ is either 0 or injective. The first isomorphism theorem then gives us further that φ is either 0 or an isomorphism, and thus every element has an inverse, hence $\mathrm{End}_R(M)$ has the structure of a division ring. □

Sub problems for this lemma:

1. Prove that the kernel of a homomorphism is an ideal.
2. Prove the first isomorphism theorem for rings.

Lemma 2. Let S be a simple R module, $D = \mathrm{End}_R(S)$ and $M = \bigoplus_{j=1}^{n} S$. Then, $\mathrm{End}_R(M) = M_n(D)$.

Proof. This is Theorem 4.4.1 in [1], combined with Schur's lemma Corollary 4.3.8, which tells us that the endomorphism ring is just the product of the endomorphism rings of the simple components. □

This of course, has the sub-problem of proving the corollary, which is straightforward but a little long to write out.

We then intend to break the proof of the main theorem down into 4 smaller steps:

Lemma 3. R has finitely many non-isomorphic simple right modules S_i

Proof. Since R is semisimple, we know that $R \cong \bigoplus_{i \in I} S_i$ for some indexing set I and $S_i \cong I_i^{n_i}$, for I mutually non-isomorphic minimal left ideals. Since R is Artinian, it satisfies the descending chain condition, and so I must be finite. □

Lemma 4. For any ring R, we have that $\mathrm{End}_R(R_R) \cong R$.

Proof. Take the 'obvious' map $\phi_r(s) : R \to \mathrm{End_R(R_R)}$ by $s \mapsto rs$. It's straightforward to show this is an isomorphism. □

Both of these should be pretty direct I think.

Lemma 5. For an Artinian semi-simple right R module,

$$\mathrm{End}_R(M) \cong \bigoplus_{i=1}^{m} M_{a_i}(D_i)$$

for division rings D_i and non-negative integers a_i.

Proof. The proof more or less consists of applying Lemmas 1-3. □

Proof (Artin-Wedderburn). The final proof of the Artin-Wedderburn theorem is then achieved by applying Lemma 4 and Lemma 5. □

If this turns out to be easier than expected, we would like to extend this result to rngs (rings without unity), however this approach would no longer work, and we would instead need to follow an argument in Hungerford. We do not expect to get to this point.

Imports

Initially, we would like to keep imports minimal, importing as a rough guide the following:

1. Algebra.Field.Defs provides basic definitions, such as the division ring;
2. Algebra.Module.Basic, Algebra.Ring.Basic and Data.Matrix.Basic for elementary definitions and properties;
3. Algebra.Ring.Equiv to equip us to treat equivalence of the relevant mathematical objects;
4. Algebra.DirectSum.Module to take direct sums of modules.

Note that instead of importing for example, Algebra.Module.Basic, we could import the more rudimentary Algebra.Module.Defs, but this would leave us needing to prove many elementary properties. We would consider doing this only if the project were to move more quickly than expected. On the other hand, if the project proves more cumbersome than expected then we will import other modules. Some helpful examples would be RingTheory.FiniteLength and selected submodules from RingTheory.SimpleModule.

Workload Split

We expect that working from the definitions, lemma 1 and lemma 2 should be quite a substantial amount of work, and so we would have one person work on each of these. The third person could then focus on the main proof, which involves lemma 3 and 4, then writing the code to bring the previous lemmas together. That being said we are still unsure of the exact breakdown, and very ready to redistribute this work as the project develops. An outline of the intended workload split is given below.

Ross

1. Prove that for $M = \oplus_i S_i$ we have $\mathrm{End}_R(M) = \prod \mathrm{End}_R(S_i)$. I expect this to be somewhat involved.
2. Use this together with Schur's lemma to prove lemma 2.
3. Help with the 'main proof' and bringing everything together.

Joseph

1. Formalise the result that the kernel of a homomorphism is an ideal.
2. Formalise the first isomorphism theorem for rings.

3. Use these results to formalise Schur's Lemma.
4. Assist with the main proof if the above are complete.

Jiehui

1. Lemma 3,4,5.
2. If possible, generalization of Lemma 2 to additive category.
3. If possible, generalization of the everything to rng-with-operator.

References

[1] Basic Algebra, Cohn, Paul M, Springer, nov 2012, pg138-139