Sophie Germain's Theorem

Benoît Ballenghien
Université Paris-Saclay, CNRS, ENS Paris-Saclay, LMF

February 6, 2026

Contents

1 Introduction

Fermat's Last Theorem (often abbreviated to FLT) states that for any integer $2 < n$, the equation $x^n + y^n = z^n$ has no nontrivial solution in the integers. Pierre de Fermat first conjectured this result in the 17th century, claiming to have a proof that did not fit in the margin of his notebook. However, it remained an open problem for centuries until Andrew Wiles and Richard Taylor provided a complete proof in 1995 using advanced techniques from algebraic geometry and modular forms.

But in the meantime, many mathematicians have made partial progress on the problem. In particular, Sophie Germain's theorem states that p is a prime such that $2 * p + 1$ is also a prime, then there are no integer solutions to the equation $x^p + y^p = z^p$ such that p divides neither x, y nor z.

This result is not only included in the extended list of Freek's "Top 100 theorems" [1], but is also very familiar to students taking the French "agrégation" mathematics competitive examination. Hoping that this submission might also be useful to them, we developed separately the classical version of the theorem as presented in [1] and a generalization that one can find for example in [2].

[1] http://www.cs.ru.nl/~freek/100/

The session displayed in 1 is organized as follows:

- `FLT_Sufficient_Conditions` provides sufficient conditions for proving FLT,
- `SG_Premilinaries` establish some useful lemmas and introduces the concept of Sophie Germain prime,
- `SG_Theorem` proves Sophie Germain's theorem and
- `SG_Generalization` gives a generalization of it.

2 Preliminaries

2.1 Coprimality

We start with this useful elimination rule: when *a* and *b* are not *coprime* and are not both equal to *0*, there exists some common *prime* factor.

lemma (**in** *factorial-semiring-gcd*) *not-coprime-nonzeroE* :
‹⟦¬ *coprime a b*; *a* ≠ *0* ∨ *b* ≠ *0*; ⋀*p*. *prime p* ⟹ *p dvd a* ⟹ *p dvd b* ⟹ *thesis*⟧ ⟹ *thesis*›
⟨*proof*⟩

Still referring to the notion of *coprime* (but generalized to a set), we prove that when *Gcd A* ≠ *0*, the elements of {*a div Gcd A* |*a*. *a* ∈ *A*} are setwise *coprime*.

lemma (**in** *semiring-Gcd*) *GCD-div-Gcd-is-one* :
‹(*GCD a* ∈ *A*. *a div Gcd A*) = *1*› **if** ‹*Gcd A* ≠ *0*›
⟨*proof*⟩

2.2 Power

Now we want to characterize the fact of admitting an *n*-th root with a condition on the *multiplicity* of each prime factor.

lemma *exists-nth-root-iff* :
‹(∃ *x*. *normalize y* = *x* ^ *n*) ⟷ (∀ *p*∈*prime-factors y*. *n dvd multiplicity p y*)›
if ‹*y* ≠ *0*› **for** *y* :: ‹'*a* :: *factorial-semiring-multiplicative*›
⟨*proof*⟩

We use this result to obtain the following elimination rule.

corollary *prod-is-some-powerE* :
fixes *a b* :: ‹'*a* :: *factorial-semiring-multiplicative*›

assumes ‹*coprime a b*› **and** ‹$a * b = x ^ n$›
obtains α **where** ‹*normalize* $a = \alpha ^ n$›
⟨*proof*⟩

2.3 Sophie Germain Prime

Finally, we introduce Sophie Germain primes.

definition *SophGer-prime* :: ‹*nat* ⇒ *bool*› (‹*SG*›)
where ‹*SG p* ≡ *odd p* ∧ *prime p* ∧ *prime* (*2* ∗ *p* + *1*)›

lemma *SophGer-primeI* : ‹*odd p* ⟹ *prime p* ⟹ *prime* (*2* ∗ *p* + *1*) ⟹ *SG p*›
⟨*proof*⟩

lemma *SophGer-primeD* : ‹*odd p*› ‹*prime p*› ‹*prime* (*2* ∗ *p* + *1*)› **if** ‹*SG p*›
⟨*proof*⟩

We can easily compute Sophie Germain primes less than *2000*.

value ‹[*p*. *p* ← [*0*..*2000*], *SG* (*nat p*)]›

context fixes *p* **assumes** ‹*SG p*› **begin**

⟨*ML*⟩

lemma *nonzero* : ‹$p \neq 0$› ⟨*proof*⟩

lemma *pos* : ‹$0 < p$› ⟨*proof*⟩

lemma *ge-3* : ‹$3 \leq p$›
⟨*proof*⟩

lemma *ge-7* : ‹$7 \leq 2 * p + 1$› ⟨*proof*⟩

lemma *notcong-zero* :
‹[− *3* ≠ *0* :: *int*] (*mod 2* ∗ *p* + *1*)› ‹[− *1* ≠ *0* :: *int*] (*mod 2* ∗ *p* + *1*)›
‹[*1* ≠ *0* :: *int*] (*mod 2* ∗ *p* + *1*)› ‹[*3* ≠ *0* :: *int*] (*mod 2* ∗ *p* + *1*)›
⟨*proof*⟩

lemma *notcong-p* :
‹[− *1* ≠ *p* :: *int*] (*mod 2* ∗ *p* + *1*)›
‹[*0* ≠ *p* :: *int*] (*mod 2* ∗ *p* + *1*)›
‹[*1* ≠ *p* :: *int*] (*mod 2* ∗ *p* + *1*)›
⟨*proof*⟩

lemma *p-th-power-mod-q* :
‹[$m ^ p = 1$] (*mod 2* ∗ *p* + *1*) ∨ [$m ^ p = - 1$] (*mod 2* ∗ *p* + *1*)›

if ‹$\neg$ *2* $*$ *p* + *1* *dvd* *m*› **for** *m* :: *int*
⟨*proof*⟩

end

2.4 Fermat's little Theorem for Integers

lemma *fermat-theorem-int* :
‹[*a* ^ (*p* − *1*) = *1*] (*mod p*)› **if** ‹*prime p*› **and** ‹$\neg$ *p dvd a*›
for *p* :: *nat* **and** *a* :: *int*
⟨*proof*⟩

3 Sufficient Conditions for FLT

Recall that FLT stands for "Fermat's Last Theorem". FLT states that there is no nontrivial integer solutions to the equation $x^n + y^n = z^n$ for any natural number $2 < n$. as soon as the natural number n is greater than 2. More formally: $2 < n \Longrightarrow \nexists x\ y\ z.\ x^n + y^n = z^n$. We give here some sufficient conditions.

3.1 Coprimality

We first notice that it is sufficient to prove FLT for integers x, y and z that are (setwise) *coprime*.

lemma (**in** *semiring-Gcd*) *FLT-setwise-coprime-reduction* :
assumes ‹*x* ^ *n* + *y* ^ *n* = *z* ^ *n*› ‹*x* ≠ *0*› ‹*y* ≠ *0*› ‹*z* ≠ *0*›
defines ‹*d* ≡ *Gcd* {*x*, *y*, *z*}›
shows ‹(*x div d*) ^ *n* + (*y div d*) ^ *n* = (*z div d*) ^ *n*› ‹*x div d* ≠ *0*›
‹*y div d* ≠ *0*› ‹*z div d* ≠ *0*› ‹*Gcd* {*x div d*, *y div d*, *z div d*} = *1*›
⟨*proof*⟩

corollary (**in** *semiring-Gcd*) *FLT-for-coprime-is-sufficient* :
‹∄*x y z*. *x* ≠ *0* ∧ *y* ≠ *0* ∧ *z* ≠ *0* ∧ *Gcd* {*x*, *y*, *z*} = *1* ∧ *x* ^ *n* + *y* ^ *n* = *z* ^ *n*
⟹
∄*x y z*. *x* ≠ *0* ∧ *y* ≠ *0* ∧ *z* ≠ *0* ∧ *x* ^ *n* + *y* ^ *n* = *z* ^ *n*›
⟨*proof*⟩
lemma ‹*OFCLASS*(*int*, *semiring-Gcd-class*)› ⟨*proof*⟩

This version involving congruences will be useful later.

lemma *FLT-setwise-coprime-reduction-mod-version* :
fixes *x y z* :: *int*

assumes ‹x ^ n + y ^ n = z ^ n› ‹[$x \neq 0$] (mod m)› ‹[$y \neq 0$] (mod m)› ‹[$z \neq 0$] (mod m)›
defines ‹$d \equiv Gcd$ {x, y, z}›
shows ‹(x div d) ^ n + (y div d) ^ n = (z div d) ^ n› ‹[x div $d \neq 0$] (mod m)› ‹[y div $d \neq 0$] (mod m)› ‹[z div $d \neq 0$] (mod m)› ‹Gcd {x div d, y div d, z div d} = 1›
⟨*proof*⟩

Actually, it is sufficient to prove FLT for integers x, y and z that are pairwise *coprime*

lemma (**in** *semiring-Gcd*) *FLT-setwise-coprime-imp-pairwise-coprime* :
‹*coprime* x y› **if** ‹$n \neq 0$› ‹x ^ n + y ^ n = z ^ n› ‹Gcd {x, y, z} = 1›
⟨*proof*⟩

3.2 Odd prime Exponents

From `Fermat3_4`, FLT is already proven for $n = 4$. Using this, we can prove that it is sufficient to prove FLT for *odd prime* exponents.

lemma (**in** *semiring-1-no-zero-divisors*) *FLT-exponent-reduction* :
assumes ‹x ^ n + y ^ n = z ^ n› ‹$x \neq 0$› ‹$y \neq 0$› ‹$z \neq 0$› ‹p dvd n›
shows ‹(x ^ (n div p)) ^ p + (y ^ (n div p)) ^ p = (z ^ (n div p)) ^ p›
‹x ^ (n div p) $\neq 0$› ‹y ^ (n div p) $\neq 0$› ‹z ^ (n div p) $\neq 0$›
⟨*proof*⟩

lemma ‹*OFCLASS*(*int*, *semiring-1-no-zero-divisors-class*)› ⟨*proof*⟩

lemma *odd-prime-or-four-factorE* :
fixes n :: *nat* **assumes** ‹$2 < n$›
obtains p **where** ‹p dvd n› ‹*odd* p› ‹*prime* p› | ‹4 dvd n›
⟨*proof*⟩

Finally, proving FLT for odd prime exponents is sufficient.

corollary *FLT-for-odd-prime-exponents-is-sufficient* :
‹$\nexists$ x y z :: *int*. $x \neq 0 \wedge y \neq 0 \wedge z \neq 0 \wedge$ x ^ n + y ^ n = z ^ n› **if** ‹$2 < n$›
and *odd-prime-FLT* :
‹$\bigwedge p$. *odd* $p \Longrightarrow$ *prime* $p \Longrightarrow$
$\nexists$ x y z :: *int*. $x \neq 0 \wedge y \neq 0 \wedge z \neq 0 \wedge$ x ^ p + y ^ p = z ^ p›
⟨*proof*⟩

4 Sophie Germain's Theorem: classical Version

The proof we give here is from [1].

4.1 A Crucial Lemma

lemma *Sophie-Germain-lemma-computation* :
fixes $x\ y :: int$ **assumes** ‹$odd\ p$›
defines ‹$S \equiv \sum k = 0..p - 1.\ (- y) \,\hat{}\, (p - 1 - k) * x \,\hat{}\, k$›
shows ‹$(x + y) * S = x \,\hat{}\, p + y \,\hat{}\, p$›
⟨*proof*⟩

lemma *Sophie-Germain-lemma-computation-cong-simp* :
fixes $p :: nat$ **and** $n\ x\ y :: int$ **assumes** ‹$p \neq 0$› ‹$[y = - x]\ (mod\ n)$›
defines ‹$S \equiv \lambda x\ y.\ \sum k = 0..p - 1.\ (- y) \,\hat{}\, (p - 1 - k) * x \,\hat{}\, k$›
shows ‹$[S\ x\ y = p * x \,\hat{}\, (p - 1)]\ (mod\ n)$›
⟨*proof*⟩

lemma *Sophie-Germain-lemma-only-possible-prime-common-divisor* :
fixes $x\ y\ z :: int$ **and** $p :: nat$
defines *S-def*: ‹$S \equiv \lambda x\ y.\ \sum k = 0..p - 1.\ (- y) \,\hat{}\, (p - 1 - k) * x \,\hat{}\, k$›
assumes ‹$prime\ p$› ‹$prime\ q$› ‹$coprime\ x\ y$› ‹$q\ dvd\ x + y$› ‹$q\ dvd\ S\ x\ y$›
shows ‹$q = p$›
⟨*proof*⟩

lemma *Sophie-Germain-lemma* :
fixes $x\ y\ z :: int$
assumes ‹$odd\ p$› **and** ‹$prime\ p$› **and** *fermat* : ‹$x \,\hat{}\, p + y \,\hat{}\, p + z \,\hat{}\, p = 0$›
and ‹$[x \neq 0]\ (mod\ p)$› **and** ‹$coprime\ y\ z$›
defines ‹$S \equiv \sum k = 0..p - 1.\ (- z) \,\hat{}\, (p - 1 - k) * y \,\hat{}\, k$›
shows ‹$\exists\, a\ \alpha.\ y + z = a \,\hat{}\, p \wedge S = \alpha \,\hat{}\, p$›
⟨*proof*⟩

4.2 The Theorem

theorem *Sophie-Germain-theorem* :
‹$\nexists\, x\ y\ z :: int.\ x \,\hat{}\, p + y \,\hat{}\, p = z \,\hat{}\, p \wedge [x \neq 0]\ (mod\ p) \wedge$
$[y \neq 0]\ (mod\ p) \wedge [z \neq 0]\ (mod\ p)$› **if** *SG* : ‹$SG\ p$›
⟨*proof*⟩

5 Sophie Germain's Theorem: generalized Version

The proof we give here is from [2].

5.1 Auxiliary Primes

abbreviation *non-consecutivity-condition* :: ‹$nat \Rightarrow nat \Rightarrow bool$› (‹$NC$›)
where ‹$NC\ p\ q \equiv \nexists\, x\ y :: int.\ [x \neq 0]\ (mod\ q) \wedge [y \neq 0]\ (mod\ q) \wedge [x \,\hat{}\, p = 1 + y \,\hat{}\, p]\ (mod\ q)$›

lemma *non-consecutivity-condition-bis* :
‹*NC p q* ⟷ (∄ *x y a b*. [*a* :: *int* ≠ *0*] (*mod q*) ∧ [*a* ^ *p* = *x*] (*mod q*) ∧
[*b* :: *int* ≠ *0*] (*mod q*) ∧ [*b* ^ *p* = *y*] (*mod q*) ∧ [*x* = *1* + *y*] (*mod q*))›
⟨*proof*⟩

abbreviation *not-pth-power* :: ‹*nat* ⇒ *nat* ⇒ *bool*› (‹*PPP*›)
where ‹*PPP p q* ≡ ∄ *x* :: *int*. [*p* = *x* ^ *p*] (*mod q*)›

definition *auxiliary-prime* :: ‹*nat* ⇒ *nat* ⇒ *bool*› (‹*aux′-prime*›)
where ‹*aux-prime p q* ≡ *prime p* ∧ *prime q* ∧ *NC p q* ∧ *PPP p q*›

lemma *auxiliary-primeI* :
‹⟦*prime p*; *prime q*; *NC p q*; *PPP p q*⟧ ⟹ *aux-prime p q*›
⟨*proof*⟩

lemma *auxiliary-primeD* :
‹*prime p*› ‹*prime q*› ‹*NC p q*› ‹*PPP p q*› **if** ‹*aux-prime p q*›
⟨*proof*⟩

We do not give code equation yet, let us first work around these notions.

lemma *gen-mult-group-mod-prime-as-ord* : ‹*ord p g* = *p* − *1*›
if ‹*prime p*› ‹{*1* .. *p* − *1*} = {*g* ^ *k mod p* |*k*. *k* ∈ *UNIV*}›
⟨*proof*⟩

lemma *exists-nth-power-mod-prime-iff* :
fixes *p n* **assumes** ‹*prime p*›
defines *d-def* : ‹*d* ≡ *gcd n* (*p* − *1*)›
shows ‹(∃ *x* :: *int*. [*a* = *x* ^ *n*] (*mod p*)) ⟷
(*n* ≠ *0* ∧ [*a* = *0*] (*mod p*)) ∨ [*a* ^ ((*p* − *1*) *div d*) = *1*] (*mod p*)›
⟨*proof*⟩

corollary *not-pth-power-iff* :
‹*PPP p q* ⟷ [*p* ≠ *0*] (*mod q*) ∧ [*p* ^ ((*q* − *1*) *div gcd p* (*q* − *1*)) ≠ *1*] (*mod q*)›
if ‹*prime p*› ‹*prime q*›
⟨*proof*⟩

corollary *not-pth-power-iff-mod* :
‹*PPP p q* ⟷ ¬ *q dvd p* ∧ *p* ^ ((*q* − *1*) *div gcd p* (*q* − *1*)) *mod q* ≠ *1*›
if ‹*prime p*› **and** ‹*prime q*›
⟨*proof*⟩

lemma *non-consecutivity-condition-iff-enum-mod* :

— This version is oriented towards code generation.
‹$NC\ p\ q \longleftrightarrow$
$(\forall x \in \{1..q - 1\}.\ let\ x\text{-}p\text{-}mod = x \ \hat{}\ p\ mod\ q$
$in\ \forall y \in \{1..q - 1\}.\ x\text{-}p\text{-}mod \neq (1 + y \ \hat{}\ p\ mod\ q)\ mod\ q)$›
if ‹$q \neq 0$›
⟨*proof*⟩

lemma *auxiliary-prime-iff-enum-mod* [*code*] :
— We will have a more optimized version later.
‹$aux\text{-}prime\ p\ q \longleftrightarrow$
$prime\ p \wedge prime\ q \wedge$
$\neg\ q\ dvd\ p \wedge p \ \hat{}\ ((q - 1)\ div\ gcd\ p\ (q - 1))\ mod\ q \neq 1 \wedge$
$(\forall x \in \{1..q - 1\}.\ let\ x\text{-}p\text{-}mod = x \ \hat{}\ p\ mod\ q$
$in\ \forall y \in \{1..q - 1\}.\ x\text{-}p\text{-}mod \neq (1 + y \ \hat{}\ p\ mod\ q)\ mod\ q)$›
⟨*proof*⟩

We can for example compute pairs of auxiliary primes less than *110*.

value ‹$[(p, q).\ p \leftarrow [1..110],\ q \leftarrow [1..110],\ aux\text{-}prime\ (nat\ p)\ (nat\ q)]$›

lemma *auxiliary-primeI′* :
‹$[\![prime\ p;\ prime\ q;\ \neg\ q\ dvd\ p;\ p \ \hat{}\ ((q - 1)\ div\ gcd\ p\ (q - 1))\ mod\ q \neq 1;$
$\bigwedge x\ y.\ x \in \{1..q - 1\} \Longrightarrow y \in \{1..q - 1\} \Longrightarrow [x \ \hat{}\ p \neq 1 + y \ \hat{}\ p]\ (mod\ q)]\!]$
$\Longrightarrow aux\text{-}prime\ p\ q$›
⟨*proof*⟩

lemma *two-is-not-auxiliary-prime* : ‹$\neg\ aux\text{-}prime\ p\ 2$›
⟨*proof*⟩

lemma *auxiliary-prime-of-2* : ‹$aux\text{-}prime\ 2\ q \longleftrightarrow q = 3 \vee q = 5$›
⟨*proof*⟩

An auxiliary prime q of p is generally of the form $q = (2::'a) * n * p + 1$.

lemma *auxiliary-prime-pattern-aux* :
‹$\exists x\ y.\ [x \neq 0]\ (mod\ q) \wedge [y \neq 0]\ (mod\ q) \wedge [x \ \hat{}\ p = 1 + y \ \hat{}\ p]\ (mod\ q)$›
if ‹$p \neq 0$› ‹$prime\ q$› ‹$coprime\ p\ (q - 1)$› ‹$odd\ q$›
⟨*proof*⟩

theorem *auxiliary-prime-pattern* :
‹$p = 2 \wedge (q = 3 \vee q = 5) \vee odd\ p \wedge (\exists n \geq 1.\ q = 2 * n * p + 1)$› **if** *aux-p* :
‹$aux\text{-}prime\ p\ q$›
⟨*proof*⟩

lemma *auxiliary-prime-imp-less* : ‹*aux-prime p q* $\Longrightarrow$ *p* < *q*›
⟨*proof*⟩

lemma *auxiliary-primeE* :
assumes ‹*aux-prime p q*›
obtains ‹*p* = *2*› ‹*q* = *3*› | ‹*p* = *2*› ‹*q* = *5*›
| *n* **where** ‹*odd p*› ‹*1* ≤ *n*› ‹*q* = *2* ∗ *n* ∗ *p* + *1*›
‹*NC p* (*2* ∗ *n* ∗ *p* + *1*)› ‹*PPP p* (*2* ∗ *n* ∗ *p* + *1*)›
⟨*proof*⟩

With this, we can quickly eliminate numbers that cannot be auxiliary.

declare *auxiliary-prime-iff-enum-mod* [*code del*]

lemma *auxiliary-prime-iff-enum-mod-optimized* [*code*] :
‹*aux-prime p q* ⟷
p = *2* ∧ (*q* = *3* ∨ *q* = *5*) ∨
p < *q* ∧
2 ∗ *p dvd q* − *1* ∧
prime p ∧ *prime q* ∧
¬ *q dvd p* ∧ *p* ^ ((*q* − *1*) *div gcd p* (*q* − *1*)) *mod q* ≠ *1* ∧
(∀ *x* ∈ {*1*..*q* − *1*}. *let x-p-mod* = *x* ^ *p mod q*
in ∀ *y* ∈ {*1*..*q* − *1*}. *x-p-mod* ≠ (*1* + *y* ^ *p mod q*) *mod q*)›
⟨*proof*⟩

value ‹[(*p*, *q*). *p* ← [*1*..*1000*], *q* ← [*1*..*110*], *aux-prime* (*nat p*) (*nat q*)]›

5.2 Sophie Germain Primes are auxiliary

When *p* is an *odd prime* and *2* ∗ *p* + *1* is also a *prime* (what we call a Sophie Germain *prime*), *2* ∗ *p* + *1* is automatically an *aux-prime*.

lemma *SophGer-prime-imp-auxiliary-prime* :
fixes *p* **assumes** ‹*SG p*› **defines** *q-def*: ‹*q* ≡ *2* ∗ *p* + *1*›
shows ‹*aux-prime p q*›
⟨*proof*⟩

5.3 Main Theorems

theorem *Sophie-Germain-auxiliary-prime* :
‹*q dvd x* ∨ *q dvd y* ∨ *q dvd z*›
if ‹*x* ^ *p* + *y* ^ *p* = *z* ^ *p*› **and** ‹*aux-prime p q*› **for** *x y z* :: *int*
⟨*proof*⟩

theorem *Sophie-Germain-generalization* :
‹∄ *x y z* :: *int*. *x* ^ *p* + *y* ^ *p* = *z* ^ *p* ∧
[*x* ≠ *0*] (*mod* p^2) ∧ [*y* ≠ *0*] (*mod* p^2) ∧ [*z* ≠ *0*] (*mod* p^2)›
if *odd-p* : ‹*odd p*› **and** *aux-prime* : ‹*aux-prime p q*›

⟨*proof*⟩

Since $SG\ p \Longrightarrow aux\text{-}prime\ p\ (2 * p + 1)$, this result is a generalization of $SG\ p \Longrightarrow \nexists x\ y\ z.\ x^p + y^p = z^p \wedge [x \neq 0]\ (mod\ int\ p) \wedge [y \neq 0]\ (mod\ int\ p) \wedge [z \neq 0]\ (mod\ int\ p)$.

References

[1] S. Francinou, H. Gianella, and S. Nicolas. *Oraux X-ENS Algèbre 1*. Cassini, 2014.

[2] A. Kiefer. Le théorème de Fermat vu par M. Le Blanc. *Brussels Summer School of Mathematics, Notes de la cinquième BSSM*, 2012.

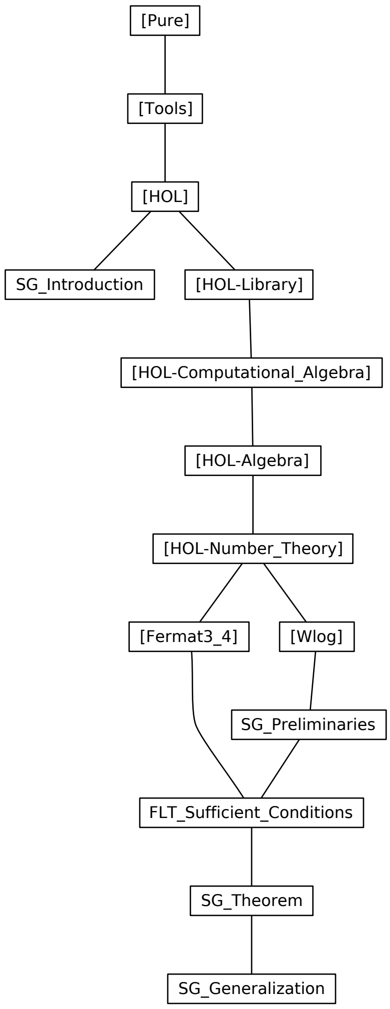

Figure 1: Dependency graph of the session `Sophie_Germain`