# A Formalisation of Lehmer's Primality Criterion 

By Simon Wimmer and Lars Noschinski

September 13, 2023


#### Abstract

In 1927, Lehmer presented criterions for primality, based on the converse of Fermat's litte theorem [2]. This work formalizes the second criterion from Lehmer's paper, a necessary and sufficient condition for primality.

As a side product we formalize some properties of Euler's $\varphi$-function, the notion of the order of an element of a group, and the cyclicity of the multiplicative group of a finite field.


## Contents

## 1 Introduction

## 1 Introduction

Section ?? provides some technical lemmas about polynomials. Section ?? to ?? formalize some basic number-theoretic and algebraic properties: Euler's $\varphi$-function, the order of an element of a group and an upper bound of the number of roots of a polynomial. Section ?? combines these results to prove that the multiplicative group of a finite field is cyclic. Based on that, Section 2 formalizes an extended version of Lehmer's Theorem, which gives us necessary and sufficient conditions to decide whether a number is prime.
theory Lehmer
imports
Main
HOL-Number-Theory.Residues
begin

## 2 Lehmer's Theorem

In this section we prove Lehmer's Theorem [2] and its converse. These two theorems characterize a necessary and complete criterion for primality. This
criterion is the basis of the Lucas-Lehmer primality test and the primality certificates of Pratt [3].

```
lemma mod-1-coprime-nat:
    coprime \(a b\) if \(0<n\left[a^{\wedge} n=1\right](\bmod b)\) for \(a b::\) nat
proof -
    from that coprime-1-left have coprime \(\left(a^{\wedge} n\right) b\)
        using cong-imp-coprime cong-sym by blast
    with \(\langle 0<n\rangle\) show ?thesis
        by \(\operatorname{simp}\)
qed
```

This is a weak variant of Lehmer's theorem: All numbers less then $p-1$ must be considered.

```
lemma lehmers-weak-theorem:
    assumes \(2 \leq p\)
    assumes min-cong1: \(\bigwedge x .0<x \Longrightarrow x<p-1 \Longrightarrow\left[a^{\wedge} x \neq 1\right](\bmod p)\)
    assumes cong1: \(\left[a^{\wedge}(p-1)=1\right](\bmod p)\)
    shows prime \(p\)
proof (rule totient-imp-prime)
    from \(\langle 2 \leq p\rangle\) cong1 have coprime a \(p\)
        by (intro mod-1-coprime-nat \([\) of \(p-1]\) ) auto
    then have \(\left[a^{\wedge}\right.\) totient \(\left.p=1\right](\bmod p)\)
        by (intro euler-theorem) auto
    then have totient \(p \geq p-1 \vee\) totient \(p=0\)
        using min-cong1 [of totient \(p\) ] by fastforce
    moreover have totient \(p>0\)
        using \(\langle 2 \leq p\rangle\) by \(\operatorname{simp}\)
    moreover from \(\langle p \geq 2\rangle\) have totient \(p<p\) by (intro totient-less) auto
    ultimately show totient \(p=p-1\) by presburger
qed (insert \(\langle p \geq 2\rangle\), auto)
lemma prime-factors-elem:
    fixes \(n::\) nat assumes \(1<n\) shows \(\exists p\). \(p \in\) prime-factors \(n\)
    using assms by (cases prime n) (auto simp: prime-factors-dvd prime-factor-nat)
lemma cong-pow-1-nat:
    \(\left[a^{\wedge} x=1\right](\bmod b)\) if \([a=1](\bmod b)\) for \(a b::\) nat
    using cong-pow [of a \(1 \quad b x]\) that by simp
lemma cong-gcd-eq-1-nat:
    fixes \(a b:\) nat
    assumes \(0<m\) and cong-props: \(\left[a^{\wedge} m=1\right](\bmod b)\left[a^{\wedge} n=1\right](\bmod b)\)
    shows \([a \wedge g c d m n=1](\bmod b)\)
proof -
    obtain \(c d\) where \(g c d: m * c=n * d+g c d m n\) using bezout-nat[of \(m n]<0\)
\(<m\) >
        by auto
    have cong-m: \(\left[a^{\wedge}(m * c)=1\right](\bmod b)\) and cong-n: \(\left[a^{\wedge}(n * d)=1\right](\bmod b)\)
            using cong-props by (simp-all only: cong-pow-1-nat power-mult)
```

```
    have \(\left[1 * a \wedge g c d m n=a^{\wedge}(n * d) * a \wedge g c d m n\right](\bmod b)\)
    by (rule cong-scalar-right, rule cong-sym) (fact cong-n)
    also have \([a \wedge(n * d) * a \wedge g c d m n=a \wedge(m * c)](\bmod b)\)
    using \(g c d\) by (simp add: power-add)
    also have \(\left[a^{\wedge}(m * c)=1\right](\bmod b)\) using cong-m by \(\operatorname{simp}\)
    finally show \([a \wedge\) gcd \(m n=1](\bmod b)\) by \(\operatorname{simp}\)
qed
lemma One-leq-div:
    \(1<b\) div \(a\) if \(a d v d b a<b\) for \(a b::\) nat
    using that by (metis dvd-div-mult-self mult.left-neutral mult-less-cancel2)
theorem lehmers-theorem:
    assumes \(2 \leq p\)
    assumes pf-notcong1: \(\wedge x . x \in\) prime-factors \((p-1) \Longrightarrow\left[a^{\wedge}((p-1)\right.\) div \(x)\)
\(\neq 1](\bmod p)\)
    assumes cong1: \(\left[a^{\wedge}(p-1)=1\right](\bmod p)\)
    shows prime \(p\)
proof cases
    assume \([a=1](\bmod p)\) with \(\langle 2 \leq p\rangle p f\)-notcong1 show ?thesis
        by (metis cong-pow-1-nat less-diff-conv linorder-neqE-nat linorder-not-less
            one-add-one prime-factors-elem two-is-prime-nat)
next
    assume \(A\)-notcong-1: \([a \neq 1](\bmod p)\)
    \{ fix \(x\) assume \(0<x x<p-1\)
        have \(\left[a^{\wedge} x \neq 1\right](\bmod p)\)
        proof
            assume \(\left[a{ }^{\wedge} x=1\right](\bmod p)\)
            then have \(g c d\)-cong-1: \([a \wedge \operatorname{gcd} x(p-1)=1](\bmod p)\)
                by (rule cong-gcd-eq-1-nat \([O F\langle 0<x\rangle-\operatorname{cong} 1])\)
            have \(\operatorname{gcd} x(p-1)=p-1\)
            proof (rule ccontr)
                assume \(\neg\) ?thesis
                    then have \(g c d-p 1: \operatorname{gcd} x(p-1) d v d(p-1) \operatorname{gcd} x(p-1)<p-1\)
                    using gcd-le2-nat \([\) of \(p-1 x]<2 \leq p\rangle\) by \((\) simp, linarith \()\)
                define \(c\) where \(c=(p-1)\) div \((g c d x(p-1))\)
                    then have \(p\)-1-eq: \(p-1=\operatorname{gcd} x(p-1) * c\) unfolding \(c\)-def using \(g c d-p 1\)
                by (metis dvd-mult-div-cancel)
            from \(g c d-p 1\) have \(1<c\) unfolding \(c\)-def by (rule One-leq-div)
            then obtain \(q\) where \(q\)-pf: \(q \in\) prime-factors \(c\)
                using prime-factors-elem by auto
                then have \(q d v d c\) by auto
            have \(q \in\) prime-factors \((p-1)\) using \(q-p f\langle 1<c\rangle\langle 2 \leq p\rangle\)
                by (subst p-1-eq) (simp add: prime-factors-product)
                moreover
```

```
                have [a^ ((p-1) div q)=1] (mod p)
                    by (subst p-1-eq,subst dvd-div-mult-self[OF <q dvd c>,symmetric])
                    (simp del: One-nat-def add: power-mult gcd-cong-1 cong-pow-1-nat)
        ultimately
        show False using pf-notcong1 by metis
        qed
        then show False using <x < p-1>
        by (metis <0 < x\rangle gcd-le1-nat gr-implies-not0 linorder-not-less)
    qed
}
    with lehmers-weak-theorem[OF <2 \leq p>-cong1] show ?thesis by metis
qed
```

The converse of Lehmer's theorem is also true.

```
lemma converse-lehmer-weak:
```

assumes prime-p: prime $p$
shows $\exists a \cdot\left[a^{\wedge}(p-1)=1\right](\bmod p) \wedge(\forall x .0<x \longrightarrow x \leq p-2 \longrightarrow[a \widehat{a} \neq$
1] $(\bmod p))$
$\wedge a>0 \wedge a<p$
proof -
have $p \geq 2$ by (rule prime-ge-2-nat[OF prime-p])
obtain $a$ where $a: a \in\{1 . . p-1\} \wedge\{1 . . p-1\}=\left\{a^{\wedge} i \bmod p \mid i . i \in\right.$
UNIV \}
using residue-prime-mult-group-has-gen [OF prime-p] by blast
\{
$\{$ fix $x::$ nat assume $x: 0<x \wedge x \leq p-2 \wedge[\widehat{a x}=1](\bmod p)$
have $\left\{a^{\wedge} i \bmod p \mid i . i \in U N I V\right\}=\{a \widehat{a} \bmod p \mid i .0<i \wedge i \leq x\}$
proof
show $\{a \wedge i \bmod p \mid i .0<i \wedge i \leq x\} \subseteq\{a \wedge i \bmod p \mid i . i \in U N I V\}$ by
blast
$\{$ fix $y$ assume $y: y \in\{a \widehat{i} \bmod p \mid i . i \in U N I V\}$
then obtain $i$ where $i: y=a^{\wedge} i \bmod p$ by auto
define $q r$ where $q=i$ div $x$ and $r=i \bmod x$
have $i=q * x+r$ by (simp add: $r$-def $q$-def)
hence $y-q-r: y=(((a \wedge(q * x)) \bmod p) *((a \wedge r) \bmod p)) \bmod p$
by (simp add: i power-add mod-mult-eq)
have $a{ }^{\wedge}(q * x) \bmod p=\left(a{ }^{\wedge} x \bmod p\right) \wedge q \bmod p$
by (simp add: power-mod mult.commute power-mult[symmetric])
then have $y-r: y=a \wedge r \bmod p$ using $\langle p \geq 2\rangle x$
by (simp add: cong-def $y-q-r$ )
have $y \in\left\{a^{\wedge} i \bmod p \mid i .0<i \wedge i \leq x\right\}$
proof (cases)
assume $r=0$
then have $y=\widehat{a<x \bmod p \mathbf{u s i n g}\langle p \geq 2\rangle x}$
by (simp add: cong-def $y$-r)
thus ?thesis using $x$ by blast
next
assume $r \neq 0$
thus ?thesis using $x$ by (auto simp add: $y$-r r-def)

```
            qed
        }
        thus {a^i mod p|i.i\inUNIV }\subseteq{a^ imod p | i. 0< i^ i m x by buto
    qed
    note X = this
    have p-1= card {1..p-1} by auto
    also have {1..p-1}={a^i mod p|i.1\leqi^i\leqx} using X a by auto
    also have ... =( \lambdai.a^i mod p)'{1..x} by auto
    also have card ... \leqp-2
        using Finite-Set.card-image-le[of {1..x} \lambda i. a`i mod p]x by auto
    finally have False using <2 }\leqp\rangle\mathrm{ by arith
    }
    hence }\forallx.0<x\longrightarrowx\leqp-2\longrightarrow[a^x\not=1] (\operatorname{mod}p)\mathrm{ by auto
} note a-is-gen = this
{
    assume a>1
    have }\negpdvd 
    proof (rule ccontr)
        assume }\neg\negp\mathrm{ dvd a
        hence pdvd a by auto
        have p\leqa using dvd-nat-bounds[OF - <p dvd a〉] a by simp
        thus False using {a>1\rangle a by force
    qed
    hence coprime a p
        using prime-imp-coprime-nat [OF prime-p] by (simp add: ac-simps)
    then have [a^ totient p=1] ( mod p)
        by (rule euler-theorem)
    also from prime-p have totient p=p-1
        by (rule totient-prime)
    finally have [ [ ^ (p-1) =1] ( }\operatorname{mod}p)
}
hence [a`(p-1)=1] (mod p) using a by fastforce
thus ?thesis using a-is-gen a by auto
qed
theorem converse-lehmer:
assumes prime-p:prime(p)
shows \existsa.[a`(p-1)=1] (mod p)^
    (\forallq.q\in prime-factors }(p-1)\longrightarrow[\mp@subsup{a}{}{\wedge}((p-1)\operatorname{div}q)\not=1](\operatorname{mod}p)
    \wedgea>0^a<p
proof -
    have p \geq2 by (rule prime-ge-2-nat[OF prime-p])
    obtain a where a:[a^(p-1)=1] (mod p)^(\forallx.0<x\longrightarrow
[a`x\not=1] (mod p))
    \wedge a>0^a<p
    using converse-lehmer-weak[OF prime-p] by blast
{ fix q assume q:q\in prime-factors (p-1)
    hence 0<q^q\leq p-1 using <p\geq2>
```

```
        by (auto simp add: dvd-nat-bounds prime-factors-gt-0-nat)
    hence (p-1) div q\geq1 using div-le-mono[of q p-1 q] div-self[of q] by simp
    have q\geq2 using q by (auto intro: prime-ge-2-nat)
    hence (p-1) div q< p-1 using <p\geq2\rangle by simp
    hence [a`((p-1) div q)\not=1] ( mod p) using a<(p-1) div q\geq1>
        by (auto simp add: Suc-diff-Suc less-eq-Suc-le)
}
thus ?thesis using a by auto
qed
end
```


## References

[1] K. Conrad. Cyclicity of $(\mathbf{Z} /(p))^{\times}$. http://www.math.uconn.edu/ $\sim$ kconrad/blurbs/grouptheory/cyclicFp.pdf.
[2] D. H. Lehmer. Tests for primality by the converse of fermat's theorem. Bull. Amer. Math. Soc., 33:327-340, 1927.
[3] V. R. Pratt. Every prime has a succinct certificate. SIAM Journal on Computing, 4(3):214-220, 1975.

