LECTURE NOTES ON THE COMPLEXITY OF SOME PROBLEMS IN NUMBER THEORY

Dana Angluin

Technical report 243, August 1982

Table of Contents

LECTURE NOTES ON THE COMPLEXITY OF SOME PROBLEMS IN NUMBER THEORY

Dana Angluin[1]

Department of Computer Science, Yale University

Abstract

Some basic results and algorithms from number theory are described, including the greatest common divisor, inverses, exponentiation, the theorems of Euler and Fermat, the Chinese Remainder Theorem, testing primality, the Legendre and Jacobi symbols, and finding square roots modulo primes and composites. This material is intended to provide a foundation for the study of cryptographic systems based on number theoretic problems. Some familiarity with the concepts of analysis of algorithms and complexity theory is assumed. The number theory portion is intended to be largely self-contained.

1. Introduction

These notes describe some complexity-theoretic results concerning problems of classical number theory. We assume the standard machinery of complexity theory: definitions of polynomial-time computation, the class P of sets recognizable in polynomial time, the class NP of sets recognizable nondeterministically in polynomial time, NP-completeness, and related results. On the number-theory side, the aim is to make these notes nearly self-contained, except for material on the Extended Riemann Hypothesis, occasional details, and some elementary results from group theory.

A good basic treatment of analysis of algorithms and complexity theory may be found in

[1] Preparation of these notes was partially funded by the National Science Foundation under grant number MCS-8002447. These notes were originally prepared as part of the postgraduate lectures in Artificial Intelligence and Computer Science at the University of Edinburgh.

the text of Aho, Hopcroft, and Ullman [4]. Basic results on number theory and the computational aspects of the greatest common divisor, the Chinese remainder theorem, modular arithmetic, factoring, and primality-testing may be found in the monographs of Niven and Zuckermann [12], Vinogradov [15], and Knuth [9]. The other material covered in these notes may be found in the articles of Adleman, Manders, and Miller [3], Berlekamp [5], Carmichael [6], Manders and Adleman [10], Miller [11], Pratt [13], and Solovay and Strassen [14].

2. A note on complexity theory

These notes are concerned with very broad distinctions of whether a computational problem is "easy" or "hard". It has been customary to identify "easy" with "solvable in polynomial time". This identification must be taken with a grain of salt: the nonpolynomial bound $n^{log\ log\ n}$ does not overtake the polynomial bound n^{10} until $n = 1024$. Nonetheless, the class of functions computable in polynomial time has certain desirable properties. It is closed under composition, or, to put it in computational terms, we may use polynomial time subroutines with impunity. Also, it is invariant under a variety of changes to the model of computation, so we do not have to bother about details of the model in our analyses of running times. (If you want a more definite model, assume a log-cost RAM or a multi-tape Turing machine [4].)

If polynomial time is "easy", what about "hard"? At first glance, it seems that "not computable in polynomial time" is a reasonable interpretation of "hard". There are some problems with this identification, which we consider below.

The first problem is the major embarrassment of complexity theory: the NP-complete problems. While a number of natural problems have been proved to require exponential (or more) time, the NP-complete problems are *not* among them. Nor do we have polynomial time algorithms for the NP-complete problems. What we do have is a large collection of polynomial time *reductions* showing that if problem A has a polynomial time algorithm then so does problem B. The NP-complete problems stand or fall together: if any one of them has a polynomial time algorithm, then they all do. This would not be so bad, but there are literally hundreds (see Garey and Johnson's collection [7]) of problems, some of them very important practical problems, that have been proved to be NP-complete.

There is a seductive argument, which may be termed the *argument from ignorance*, to the effect that if all these bright people have been working all this time on these very important problems without coming up with a polynomial time algorithm, then it is likely that the reason is that no such algorithm exists. (Or more succinctly: we don't know how to do it, therefore it is impossible.) The argument from ignorance is antithetical to the guiding spirit of science, and ought to be handled with great care. However, it has become customary to accept a proof of the NP-completeness of a problem as some kind of evidence of its computational "hardness".

Another problem with the identification of "hard" with "not computable in polynomial time" has to do with the purposes involved. In the customary problem setting, we imagine that our job is to compute the answer efficiently for every possible input, and an infinite collection of inputs, however sparse, where we cannot do this is synonymous with failure. In this case, "easy" means "easy everywhere", and "hard" means "hard on infinitely many inputs". However, in proposals to use "hard" problems to produce secure cryptosystems, the setting changes. In particular, we imagine that our job is to make it hard for the adversary to decrypt *any* of our encrypted messages, so that "hard" now means "hard everywhere" and "easy" means "easy somewhere". Thus, infinitely often hard is not "hard enough" in the context of cryptosystems.

One important concept that has been developed to help deal with these issues is the notion of a randomized algorithm. This is an algorithm that may call a random number generator in the course of its computation. The analysis of such an algorithm has a probabilistic aspect. There are two basic forms that this may take. The algorithm may sometimes give the wrong answer (depending on the values supplied by the random number generator) -- the analysis then bounds the probability of this kind of error. Or, the algorithm may be guaranteed to give a correct answer always, but its running time may be longer or shorter depending on the values provided by the random number generator -- in this case, the analysis bounds the expected value of the running time (or provides more information about its probability distribution). The first type of algorithm has been called a "Monte Carlo" algorithm; an example is the primality testing procedure of Solovay and Strassen [14]. The second type of algorithm has been called a "Las Vegas" algorithm; examples are the square root finding procedures of Berlekamp [5] or Adleman, Manders, and Miller [3]. Note that in neither case is there randomization over the *input* -- the probabilities depend only on the random number generator.

In the context of cryptosystems, and in a wider practical sense, it may be reasonable to take "easy" to be "computable in polynomial time by a randomized algorithm". Unfortunately there is no single reference for the basic concepts of randomized algorithms, but the papers of Gill [8] and Adleman and Manders [1, 2] may provide helpful pointers.

Most of the algorithms described in these notes have inputs and outputs that are integers. We assume that these integers are represented in binary notation (or some other notation that is related to it by polynomial time translations, e.g., decimal notation). Thus, the length of the input n is bounded by $O(log\ n)$, and a polynomial time algorithm is one that runs in time $O((log\ n)^k)$ elementary computation steps, for some constant k. All logarithms are to the base 2.

3. Some definitions, notations, facts

Z = all the integers: 0, 1, −1, 2, −2, ...

N = the natural numbers: 1, 2, 3, 4, ...

Let $a,b,d \in$ Z, $m,n,p,q \in$ N in the following.

$a \mid b \Leftrightarrow$ there exists d so that $b = a \cdot d$

$a \nmid b$ is the negation of $a \mid b$

$a \equiv b \pmod{n} \Leftrightarrow n \mid (a - b)$

$\gcd(a,b) = \max\{d\colon d \mid a$ and $d \mid b\}$, a and b not both 0.

$\mathrm{lcm}(a,b) = \min\{m\colon a \mid m$ and $b \mid m\}$, neither a nor $b = 0$.

$\mathrm{Z}_n = \{0, 1, \ldots, n-1\}$

$\mathrm{Z}_n^* = \{m\colon 1 \leq m \leq n$ and $\gcd(m,n) = 1\}$

m and n are *relatively prime* $\Leftrightarrow \gcd(m,n) = 1$

$\phi(n) = |\mathrm{Z}_n^*|$ (Euler's totient function)

p is *prime* $\Leftrightarrow p \neq 1$ and for all $1 < m < p$, $m \nmid p$

a is a *quadratic residue* mod $n \Leftrightarrow$ there exists b such that $a \equiv b^2 \pmod{n}$

a is a *quadratic nonresidue* mod $n \Leftrightarrow a$ is not a quadratic residue mod n

$\left(\frac{a}{p}\right) = +1$ if a is a quadratic residue mod n, -1 if not (Legendre symbol)

$$\left(\frac{a}{Q}\right) = \left(\frac{a}{p_1}\right)\left(\frac{a}{p_2}\right) \cdots \left(\frac{a}{p_n}\right),$$

where $Q = p_1 p_2 \cdots p_n$, each p_i is an odd prime, and $\gcd(a,Q) = 1$ (Jacobi symbol)

a is a *q−th residue* mod $n \Leftrightarrow$ the equation $x^q \equiv a \pmod{n}$ is solvable

a is a *q−th nonresidue* mod $n \Leftrightarrow a$ is not a $q-th$ residue mod n

$\lambda(n) = \mathrm{lcm}\ \{\phi(p_1^{\alpha_1}), \ldots, \phi(p_k^{\alpha_k})\}$,

where $n = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$ is the unique prime factorization of n (Carmichael's λ function)

$\lambda'(n) = \text{lcm }\{p_1 - 1, ..., p_k - 1\}$,

where $n = p_1^{\alpha_1}p_2^{\alpha_2}\cdots p_k^{\alpha_k}$ is the unique prime factorization of n (Miller's λ' function)

We note some facts.

1. $\phi(mn) = \phi(m)\phi(n)$ if $\gcd(m,n) = 1$.

2. $\phi(p^\alpha) = p^{\alpha-1}(p - 1)$ if p is prime and $\alpha \geq 1$.

3. $\phi(p) = p - 1 \Leftrightarrow p$ is prime.

4. The unique factorization theorem.

5. The division theorem: $a = bq + r$, $0 \leq r < b$.

6. The order of a subgroup of a group divides the order of the group; the order of an element of a group divides any power of the element which is equal to the identity, and other facts of elementary group theory.

4. Computing the greatest common divisor of two numbers

For this we have a 2300 year old algorithm due to Euclid (or possibly Eudoxus). Let a and b be positive integers. By repeated application of the division theorem we may find a sequence:

$$\begin{aligned}
a &= bq_1 + r_1 & 0 &< r_1 < b\\
b &= r_1q_2 + r_2 & 0 &< r_2 < r_1\\
r_1 &= r_2q_3 + r_3 & 0 &< r_3 < r_2\\
&\cdot\\
&\cdot\\
&\cdot\\
r_{k-2} &= r_{k-1}q_k + r_k & 0 &< r_k < r_{k-1}\\
r_{k-1} &= r_kq_{k+1}
\end{aligned}$$

which terminates when we find some $r_{k+1} = 0$. (Which we must do since the sequence of nonnegative integers b, r_1, r_2, ... is strictly decreasing.)

Theorem 1: $r_k = \gcd(a,b)$.

Proof: Clearly $r_k \mid r_{k-1}$, so $r_k \mid r_{k-2}$, so $r_k \mid r_{k-3}$, and so on until we find that $r_k \mid a$ and $r_k \mid b$. If d is any positive integer which divides both a and b, then $d \mid r_1$ because $r_1 = a - bq_1$, so $d \mid r_2$ since $r_2 = b - r_1q_2$, and so on until we find that $d \mid r_k$, so $d \leq r_k$. Hence $r_k = \gcd(a,b)$. □

Corollary 2: $\gcd(a,b)$ is a multiple of every common divisor of a and b.

Example 3: Find gcd(234,108), gcd(233,144).

$$\begin{aligned} 234 &= 108 \cdot 2 + 18 \\ 108 &= 18 \cdot 6 \end{aligned}$$

Hence gcd(234,108) = 18

$$\begin{aligned} 233 &= 144 \cdot 1 + 89 \\ 144 &= 89 \cdot 1 + 55 \\ 89 &= 55 \cdot 1 + 34 \\ 55 &= 34 \cdot 1 + 21 \\ 34 &= 21 \cdot 1 + 13 \\ 21 &= 13 \cdot 1 + 8 \\ 13 &= 8 \cdot 1 + 5 \\ 8 &= 5 \cdot 1 + 3 \\ 5 &= 3 \cdot 1 + 2 \\ 3 &= 2 \cdot 1 + 1 \\ 2 &= 2 \cdot 1 \end{aligned}$$

Hence gcd(233,144) = 1

It is not difficult to show that a consecutive pair of Fibonacci numbers (eg. 144 and 233) are the worst case for Euclid's algorithm in terms of the number of applications of the division theorem required for numbers of a given magnitude. Since the k*−th* Fibonacci number exceeds r^{k-2} where r is the golden ratio, we obtain:

Theorem 4: If $a,b < n$ then the number of division stages in Euclid's algorithm will be less than 1.5 log n + O(1).

This bound is an exponential improvement over the bound of b implicit in the argument above for termination; it means that Euclid's algorithm runs in time polynomial in the lengths of the binary representations of a and b.

Corollary 5: lcm(a,b) can be computed in polynomial time.

Proof: Use the identity lcm$(a,b) = a{\cdot}b/\mathrm{gcd}(a,b)$. □

Looking again at the algorithm we note that

$$r_1 = a - q_1 b$$
$$r_2 = b - q_2 r_1 = -q_2 a + (1 + q_2 q_1) b$$
$$r_3 = r_1 - q_3 r_2 = (1 + q_3 q_2) a - (q_1 + q_3 + q_3 q_2 q_1) b.$$

In general, each r_i is expressible as an integer linear combination of a and b, including the gcd r_k. We may keep track of the coefficients at each stage as follows. Initially, set $x_1 = 1$, $y_1 = -q_1$, and $x_2 = -q_2$, $y_2 = (1 + q_2 q_1)$. Then at stage i compute

$$x_i = x_{i-2} - q_i x_{i-1}$$
$$y_i = y_{i-2} - q_i y_{i-1}.$$

By induction we see that $r_i = x_i a + y_i b$ for all $i = 1,2,\ldots,k$. Thus we have proved the following theorem.

Theorem 6: There is a polynomial time algorithm which on inputs a and b computes d,x,y such that $d = \mathrm{gcd}(a,b)$ and $d = xa + yb$.

5. Multiplicative inverses in $\mathbf{Z}_n^*$

The version of Euclid's algorithm just developed is of use in computing multiplicative inverses.

Theorem 7: There is a polynomial time algorithm which on integers a and m such that $\gcd(a,m) = 1$ computes b such that $ab \equiv 1 \pmod{m}$.

Proof: By the preceding procedure find integers x and y such that

$$ax + my = \gcd(a,m) = 1.$$

Then $m \mid (ax - 1)$, so $ax \equiv 1 \pmod{m}$. Let b be x reduced modulo m. Then $ab \equiv 1 \pmod{m}$. Also any common divisor of b and m must divide 1, so $\gcd(b,m) = 1$ and $b \in \mathbf{Z}_m^*$. □

In practice of course x_i and y_i are reduced modulo m at each stage. As a consequence of the existence of multiplicative inverses in $\mathbf{Z}_m^*$ we have the following.

Theorem 8: If m is any positive integer, $\mathbf{Z}_m^*$ forms a group under multiplication modulo m.

Proof: It is not difficult to verify that $\mathbf{Z}_m^*$ is closed under multiplication modulo m. It contains the multiplicative identity 1, and by the preceding theorem contains a multiplicative inverse for each of its elements. □

Example 9: The multiplication table of $\mathbf{Z}_{15}^*$ is

	1	2	4	7	8	11	13	14
1	1	2	4	7	8	11	13	14
2	2	4	8	14	1	7	11	13
4	4	8	1	13	2	14	7	11
7	7	14	13	4	11	2	1	8
8	8	1	2	11	4	13	14	7
11	11	7	14	2	13	1	8	4
13	13	11	7	1	14	8	4	2
14	14	13	11	8	7	4	2	1

Notes:

1. The equation $x^2 \equiv 1 \pmod{15}$ has solutions $x \equiv 1,-1,4,-4 \pmod{15}$.

2. $\mathbf{Z}_{15}^*$ is isomorphic to $\mathbf{Z}_4 \times \mathbf{Z}_2$ (under addition) via $(i,j) \to (-1)^j 2^i$.

6. The theorems of Euler and Fermat

We have just seen that $\mathbf{Z}_n$ is a group under multiplication modulo n. The order of the group $\mathbf{Z}_n$ is $\phi(n)$; recalling a little group theory, we have Euler's Theorem:

Theorem 10: For all n and all a such that $\gcd(a,n) = 1$,
$a^{\phi(n)} \equiv 1 \pmod n$.

As a special case, also Fermat's Theorem:

Theorem 11: If p is prime then for all a such that $\gcd(a,p) = 1$,
$a^{p-1} \equiv 1 \pmod p$.

The converse of Fermat's Theorem would give us a test for primality if it were true. It is not true. This will be seen in the section on Carmichael Numbers.

7. The Chinese Remainder Theorem

Theorem 12: Let $m_1, m_2, \ldots, m_r$ be relatively prime in pairs. Let $m = m_1 \cdot m_2 \cdots m_r$. Let $a_1, a_2, \ldots, a_r$ be any integers. Then there is a unique $y \in \mathbf{Z}_m$ such that $y \equiv a_i \pmod{m_i}$ for $i = 1, 2, \ldots, r$.

Proof: For each i, m/m_i is an integer, call it n_i, and $\gcd(m_i, n_i) = 1$. Thus we may find an inverse b_i such that $n_i b_i \equiv 1 \pmod{m_i}$. Now let

$$y' = \sum_{i=1,r} n_i b_i a_i,$$

and let y be y' reduced modulo m. Clearly for each $i = 1, 2, \ldots, r$, since $m_i \mid n_j$ if $i \neq j$, we have

$$y \equiv y' \equiv n_i b_i a_i \pmod{m_i}.$$

If $x \in \mathbf{Z}_m$ is any other integer with the property that $x \equiv a_i \pmod{m_i}$ for each $i = 1, 2, \ldots, r$ then because the m_i's are relatively prime in pairs, $x \equiv y \pmod{m}$, so $x = y$. □

Since the key operation in the above is the computation of the multiplicative inverses b_i, we may apply previous results to get the following.

Corollary 13: There is a polynomial time procedure to compute y from $m_1, m_2, \ldots, m_r$ and $a_1, a_2, \ldots, a_r$ in the above.

Example 14: Solve the simultaneous equations

$x \equiv 13 \pmod{15}$, $x \equiv 5 \pmod{16}$, $x \equiv 4 \pmod{7}$.

Then $m = 1680$,

$n_1 = 112 \equiv 7 \pmod{15}$, and $b_1 = 13$,
$n_2 = 105 \equiv 9 \pmod{16}$, and $b_2 = 9$,
$n_3 = 240 \equiv 2 \pmod{7}$, and $b_3 = 4$.

Solution is

$x \equiv 112 \cdot 13 \cdot 13 + 105 \cdot 9 \cdot 5 + 240 \cdot 4 \cdot 4 \pmod{1680}$,
$x \equiv 613 \pmod{1680}$.

(Remark: see also the section on modular arithmetic in Knuth, Vol. II [9].)

8. Exponentiation Modulo n

Example 15: Calculate 2^{67} (mod 22).

$$\begin{aligned}
2^1 &\equiv 2\\
2^2 &\equiv 4\\
2^4 &\equiv 16\\
2^8 &\equiv 256 \equiv 14\\
2^{16} &\equiv 14\cdot 14 \equiv 20\\
2^{32} &\equiv 20\cdot 20 \equiv 4\\
2^{64} &\equiv 4\cdot 4 \equiv 16\\
2^{67} &\equiv 2^{64}\cdot 2^2\cdot 2^1 \equiv 16\cdot 4\cdot 2 \equiv 18 \pmod{22}.
\end{aligned}$$

Theorem 16: There is a polynomial time procedure which on inputs a,m,n calculates a number $y \in \mathbf{Z}_n$ such that $y \equiv a^m$ (mod n).

Proof: Suppose m is written out in binary notation as the string $b_1 b_2 \ldots b_k$, where each b_i is 0 or 1. Start with $x_0 = 1$. Inductively calculate x_{i+1} from x_i as follows.

If $b_i = 0$ then x_{i+1} is x_i^2 reduced mod n.

If $b_i = 1$ then x_{i+1} is $a\cdot x_i^2$ reduced mod n.

Output x_k.

This procedure correctly computes y, and calculates at most $2\cdot\lceil log_2\, m\rceil$ products modulo n of numbers from $\mathbf{Z}_n$, so runs in polynomial time in the lengths of a,m,n. □

(Remark: if $\gcd(a,n) = 1$ and we happen to know $\phi(n)$ then we can use Euler's Theorem to reduce m modulo $\phi(n)$ before we start this procedure.)

This theorem shows that we can exponentiate efficiently modulo n, but what about the inverse operations? Finding *roots* of numbers modulo n appears a little less tractable, and finding *logarithms* (or *indices*) of numbers modulo n seems sufficiently intractable that it has been proposed as the basis of certain schemes for cryptography. These two problems are discussed in later sections.

9. $\mathbf{Z}_p^*$ is cyclic if p is prime

Recall that for each positive integer n, $\mathbf{Z}_n^*$ is a group under multiplication modulo n. The goal of this section is to show the following theorem.

Theorem 17: If p is prime then $\mathbf{Z}_p^*$ is a cyclic group of order $p - 1$.

(At this point a certain confusion is possible: it is an elementary result of group theory that every group of prime order is cyclic, so why is this a Big Theorem? The reason is that the order of the group $\mathbf{Z}_p^*$ is $p - 1$, which is not a prime!)

Proof: Let p be a prime. Clearly, the order of $\mathbf{Z}_p^*$ is $\phi(p) = p - 1$. To see that $\mathbf{Z}_p^*$ is cyclic, we show that it has an element of order $p - 1$. This is achieved by counting elements of different orders. Let d be any positive integer such that $d \mid (p - 1)$. Define

$$S_d = \{a \in \mathbf{Z}_p^* : a \text{ is of order } d\}.$$

These sets partition $\mathbf{Z}_p^*$, so we have

$$\sum_{d|(p-1)} |S_d| = |\mathbf{Z}_p^*| = p - 1. \qquad (*)$$

Fix d such that $d \mid (p - 1)$. We show that either $|S_d| = 0$ or $|S_d| = \phi(d)$. Suppose S_d is nonempty and choose some $a \in S_d$. Then $a, a^2, \dots, a^d$ are all distinct modulo p and each one is a solution of $x^d \equiv 1 \pmod{p}$. By the lemma below, this equation has at most d solutions modulo p, so these are all of the solutions. Hence

$$S_d \subset \{a^k : 1 \leq k \leq d\}.$$

Fix k, $1 \leq k \leq d$. If $\gcd(k,d) = e > 1$, then

$$(a^k)^{d/e} = (a^{k/e})^d \equiv 1 \pmod{p},$$

so a^k has order less than d and $a^k \notin S_d$. If $\gcd(k,d) = 1$, then there exists l such that $kl \equiv 1 \pmod d$. Hence $a^{kl} \equiv a \pmod p$. For any e, $1 \leq e \leq d-1$,

$$((a^k)^e)^l \equiv a^e \not\equiv 1 \pmod p,$$

so a^k is of order d, i.e., $a^k \in S_d$.

Thus we have shown

$$S_d = \{a^k : 1 \leq k \leq d, \gcd(k,d) = 1\},$$

so $|S_d| = \phi(d)$, as required.

Now suppose that for some d such that $d \mid (p-1)$, $|S_d| = 0$. Then

$$\sum_{d|p-1} |S_d| < \sum_{d|p-1} \phi(d).$$

By the second lemma below,

$$\sum_{d|p-1} \phi(d) = p - 1,$$

which gives a contradiction with (*) above. Hence, for each d such that $d \mid (p-1)$, we have $|S_d| = \phi(d)$, so in particular, the number of elements of order $p-1$ is $\phi(p-1)$. □

Lemma 18: If p is prime and $f(x) = a_0x^n + a_1x^{n-1} + \ldots + a_n$ is such that $f(b) \not\equiv 0 \pmod p$ for some b, then $f(x) \equiv 0 \pmod p$ has at most n distinct solutions modulo p.

Proof: By induction.

i) For $n = 0$, $f(x) = a_0$, so if $f(b) \not\equiv 0 \pmod p$ for some b then $a_0 \not\equiv 0 \pmod p$ so $a_0 \equiv 0 \pmod p$ has no solutions.

ii) Assume the result for $n-1$. Let $f(x) = a_0x^n + a_1x^{n-1} + \ldots + a_n$ and let b be such that $f(b) \not\equiv 0 \pmod p$. Suppose $f(y_i) \equiv 0 \pmod p$ for $y_1, y_2, \ldots, y_{n+1}$, all distinct modulo p. If $a_0 \equiv 0 \pmod p$ then the polynomial $g(x) = a_1x^{n-1} + \ldots + a_n$ has more than $n-1$ distinct solutions modulo p, contradicting the induction hypothesis. So we may assume that $a_0 \not\equiv 0 \pmod{p}$.

Form

$$g(x) = a_0(x - y_1)(x - y_2) \cdots (x - y_n)$$

and let $h(x) = f(x) - g(x)$. Then

$$h(x) = b_0 x^{n-1} + b_1\, x^{n-2} + \dots + b_{n-1}$$

for some integers $b_1, b_2, \dots, b_{n-1}$ and $h(y_{n+1}) \not\equiv 0 \pmod{p}$. (This depends on the fact that the product of numbers that are not congruent to 0 mod p cannot be congruent to 0 mod p.) However, $h(y_i) \equiv 0 \pmod{p}$ for $y_1, y_2, \dots, y_n$ which are all distinct modulo p, contradicting the induction hypothesis.

Hence, $f(x) \equiv 0 \pmod{p}$ can have at most n solutions modulo p. □

Note that

1. $x^2 \equiv 1 \pmod{15}$ has four distinct solutions: $x \equiv 1, -1, 4, -4 \pmod{15}$.
2. $x^2 \equiv 0 \pmod{9}$ has three distinct solutions: $x \equiv 0, 3, 6 \pmod{9}$.

Lemma 19: For all positive integers n, $\sum_{d|n} \phi(d) = n$.

Proof: Let n be some positive integer. For each d such that $d \mid n$ define

$$R_d = \{\ m{\cdot}i : m = n/d \text{ and } i \in \mathbf{Z}_d^* \ \}.$$

Clearly $R_d \subset \{1,2,\dots,n\}$ and $|R_d| = \phi(d)$. Consider any $x \in \{1,2,\dots,n\}$. Let $m = \gcd(x,n)$, $d = n/m$, and $i = x/m$. Then since $x = m{\cdot}i$ and $n = m{\cdot}d$, $\gcd(i,d) = 1$, so $x \in R_d$. If for some e such that $e \mid n$ we have $x \in R_e$, then $x = m'i'$ where $n = m'e$ and $\gcd(e,i') = 1$.

Hence $m' = \text{gcd}(x,n) = m$ and $e = d$. Thus for each $x \in \{1,2,...,n\}$, x belongs to one and only one of the sets R_d. Hence

$$n = \sum_{d|n} |R_d| = \sum_{d|n} \phi(d).$$

□

Example 20: Let $n = 18$, which has divisors $d = 1, 2, 3, 6, 9, 18$.

d	m	R_d
1	18	18
2	9	9
3	6	6 12
6	3	3 15
9	2	2 4 8 10 14 16
18	1	1 5 7 11 13 17

A stronger theorem may be proved:

Theorem 21: (Niven and Zuckerman [12], p. 52) $\mathbf{Z}_n^*$ is cyclic if and only if n is 1, 2, 4, p^k, or $2p^k$ for some odd prime number p and some positive integer k.

The proof of this theorem is omitted, though we use the cyclicity of $\mathbf{Z}_{p^m}^*$ in what follows.

10. PRIMES $\in$ NP

PRIMES will denote the set of binary representations of the prime numbers, and COMPOSITES will denote the set of binary representations of the numbers that are not prime.

It is easy to see that COMPOSITES $\in$ NP: if the input is $n > 1$ then we guess two integers m and d such that $1 < m,d < n$ and then check whether $n = m{\cdot}d$. If so, we accept n.

It is not as easy to show that the complement set, PRIMES, is in NP. We follow Pratt's proof of this result [13]. First we have:

Theorem 22: n is prime if and only if $n \neq 1$ and $\mathbf{Z}_n^*$ contains an element of order $n - 1$.

Proof: Suppose n is prime. Then $n \neq 1$ and $\mathbf{Z}_n^*$ is cyclic, so $\mathbf{Z}_n^*$ contains an element of order $\phi(n) = n - 1$. Conversely, if $n \neq 1$ and $\mathbf{Z}_n^*$ contains an element of order $n - 1$, then the order of $\mathbf{Z}_n^*$ is at least $n - 1$. Then we must have $\phi(n) = n - 1$, so n is prime. □

Verifying that $g \in \mathbf{Z}_n^*$ is of order $n - 1$ entails verifying that $g^{n-1} \equiv 1 \pmod{n}$, (which can be done in polynomial time), but also that $g^i \not\equiv 1 \pmod{n}$ for all $i = 1,2,...,n - 2$. The following theorem reduces the latter work:

Theorem 23: Let $n > 1$. The element g is of order $n - 1$ in $\mathbf{Z}_n^*$ if and only if $g^{n-1} \equiv 1 \pmod{n}$ and $g^{(n-1)/p} \not\equiv 1 \pmod{n}$ for all primes p which divide $n - 1$.

Proof: The necessity of the condition is clear. Suppose $g \in \mathbf{Z}_n^*$, $g^{n-1} \equiv 1 \pmod{n}$ and

$g^{(n-1)/p} \not\equiv 1 \pmod{n}$ for all primes p dividing $n - 1$. If m is the order of g in Z_n^*, then we must have $m \mid (n - 1)$. If $m < (n - 1)$ then there exisits a prime p dividing $n - 1$ such that $m \mid (n - 1)/p$, say $(n - 1)/p = m{\cdot}d$. Thus

$$g^{(n-1)/p} \equiv (g^m)^d \equiv 1 \pmod{n}$$

contradicting the hypotheses on g. Hence $m = n - 1$ and g is of order $n - 1$ in Z_n^*. □

Thus, *if* we knew all the prime divisors of $n - 1$, we could verify that g is of order $n - 1$ in polynomial time, since at most $\lceil log\ n \rceil$ distinct primes divide $n - 1$. This may not seem like much progress; we've reduced prime testing to finding prime factorizations. However, the availability of nondeterminism now comes to the rescue.

We construct a nondeterministic procedure to test primality as follows. Given n, guess a "generator" g and a "prime factorization" of $n - 1$. (The quotes signify that this is what we would like these things to be, but they must still be checked.) Recursively check each "prime" in the "prime factorization" for primality, and then use the procedure derived from the theorem above to verify that g is of order $n - 1$ in Z_n^*. If all of this succeeds, accept n as prime -- otherwise reject n.

Example 24: Let $n = 79$.

```
1. For 79 guess g = 3 and the factorization 78 = 2·3·13
   a. Check primality of 2, 3, and 13:
      1. Recognize 2 as prime
      2. For 3 guess g = 2 and the factorization 2 = 2
         a. Check primality of 2:
            1. Recognize 2 as prime
         b. Check correctness of the order of 2 mod 3:
               $2^2 \equiv 1 \pmod{3}$
```

```
          c. Conclude that 3 is a prime
     3. For 13 guess $g = 2$ and the factorization $12 = 2^2 \cdot 3$
          a. Check primality of 2, 3:
               1. Recognize 2 as prime
               2. Recognize 3 as prime (1,a,2 above)
          b. Check correctness of the order of 2 mod 13:
                    $2^{12} \equiv 1 \pmod{13}$
                    $2^6 \not\equiv 1 \pmod{13}$
                    $2^4 \not\equiv 1 \pmod{13}$
          c. Conclude that 13 is a prime
b. Check correctness of the order of 3 mod 79:
          $3^{78} \equiv 1 \pmod{79}$
          $3^{39} \not\equiv 1 \pmod{79}$
          $3^{26} \not\equiv 1 \pmod{79}$
          $3^6 \not\equiv 1 \pmod{79}$
c. Conclude that 79 is a prime.
```

(Remark: Knuth's example [9] of the factorization of $2^{214} + 1$ may be interesting in this connection (Vol. II, p. 349).)

To verify that this procedure runs in nondeterministic polynomial time we note that the number of and total space required by all the guesses at each level of recursion are both $O(log\ n)$, and the number of levels of recursion is at most $\lceil log\ n \rceil$. The various checking operations can be done in time polynomial in $\lceil log\ n \rceil$, so we have:

Theorem 25: PRIMES $\in$ NP.

(Remark: the procedures of Miller and Solovay and Strassen, discussed in later sections, do not directly adapt to give this result because both procedures are based on finding witnesses for compositeness rather than primality.)

11. Primitive roots and indices

Let p be a prime. We have seen that $\mathbf{Z}_p^*$ is a cyclic group, and that $\mathbf{Z}_p^*$ has $\phi(p-1)$ elements of order $p-1$. Any element $g \in \mathbf{Z}_p^*$ of order of $p-1$ is called a *generator* or *primitive root* of p. Let g be a fixed primitive root of p. Each $a \in \mathbf{Z}_p^*$ has associated with it a unique integer $k \in \{0,1,...,p-1\}$ such that $a \equiv g^k \pmod p$. This k is denoted by $ind_g(a)$ and is called the *index* of a with respect to g.

Primitive roots and their associated tables of indices are useful aids to hand calculation modulo primes of moderate size, analogous to tables of logarithms, reducing multiplication and division modulo p to addition and subtraction modulo $p-1$. (See Appendix I.)

It appears to be an open problem whether or not $ind_g(a)$ can be calculated in deterministic polynomial time given a, p, g where g is guaranteed to be a primitive root of p and p is guaranteed to be prime, even assuming the Extended Riemann Hypothesis or an oracle for factoring. On the other hand, the ability to calculate indices in polynomial time doesn't seem to help much with other problems (factoring, root-finding). The apparent computational intractability of index-finding has been used as the basis of certain cryptographic schemes, for example, the proposal of Micali and Blum for the generation of cryptographically secure pseudo-random numbers (to appear in FOCS 82).

Indices are often quite useful in proofs, for example:

Theorem 26: If p is an odd prime and g is any primitive root of p then $a \in \mathbf{Z}_p^*$ has a square root modulo p if and only if $ind_g(a)$ is even.

Proof: Suppose $a \equiv g^{2m} \pmod{p}$ for some integer m. Then, letting $b \equiv g^{m} \pmod{p}$, we have $a \equiv b^2 \pmod{p}$. Conversely, suppose $a \equiv b^2 \pmod{p}$. Let $m = ind_g(b)$, so $b \equiv g^m \pmod{p}$ and $a \equiv g^{2m} \pmod{p}$. Thus

$$ind_g(a) \equiv 2m \pmod{(p-1)}.$$

Since $p - 1$ is a multiple of 2, this implies that $ind_g(a)$ is even. □

Corollary 27: If p is an odd prime, exactly half the elements of $\mathbf{Z}_p^*$ have square roots.

(Remark: This applies, *mutatis mutandis*, to q–th roots, for $q \mid (p-1)$.)

12. Quadratic residues

Recall that if p is an odd prime and a is relatively prime to p then a is a *quadratic residue* modulo p if and only if a has a square root modulo p. The Legendre symbol is defined thus:

$$\left(\frac{a}{p}\right) = \begin{array}{l} +1 \text{ if } a \text{ has a square root mod } p, \\ -1 \text{ if } a \text{ has no square root mod } p. \end{array}$$

Let g be any primitive root of p. We have shown that a has a square root modulo p if and only if $ind_g(a)$ is even. From this we obtain:

Theorem 28: For all primes p and all $a, b \in \mathbf{Z}_p^*$,

$$\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right).$$

We also note:

Theorem 29: If p is an odd prime and g is a primitive root of p then

$$g^{(p-1)/2} \equiv -1 \pmod{p}.$$

Proof: $(g^{(p-1)/2})^2 \equiv 1 \pmod{p}$. There are two distinct square roots of 1 modulo p, 1 and -1. Since g is of order $p-1$, $g^{(p-1)/2} \not\equiv 1 \pmod{p}$. □

In turn we get a way of calculating the Legendre symbol:

Theorem 30: If p is an odd prime and a is relatively prime to p then

$$a^{(p-1)/2} \equiv \left(\frac{a}{p}\right) \pmod{p}.$$

Proof: Suppose $\left(\frac{a}{p}\right) = 1$. Then there exists $b \in \mathbf{Z}_p^*$ such that $a \equiv b^2 \pmod{p}$. Hence

$$a^{(p-1)/2} \equiv b^{p-1} \equiv 1 \equiv \left(\frac{a}{p}\right) \pmod{p}.$$

If $\left(\frac{a}{p}\right) = -1$ and g is any primitive root of p, we have $a \equiv g^{2m+1} \pmod{p}$ for some integer m. Hence

$$a^{(p-1)/2} \equiv g^{m(p-1)+(p-1)/2} \equiv g^{(p-1)/2} \equiv -1 \pmod{p}.$$

□

This theorem gives us one method of calculating the Legendre symbol of a with respect to p in polynomial time; the section on the Jacobi symbol gives another.

13. The Jacobi symbol

The Jacobi symbol generalizes the Legendre symbol, but not in the respect of indicating the existence of square roots.

If Q is an odd number greater than 1, and $Q = p_1 \cdot p_2 \cdots p_k$ where each p_i is prime, and a is relatively prime to Q then the Jacobi symbol is defined in terms of the Legendre symbol as follows:

$$\left(\frac{a}{Q}\right) = \left(\frac{a}{p_1}\right)\left(\frac{a}{p_2}\right) \cdots \left(\frac{a}{p_k}\right).$$

Example 31: $\left(\frac{2}{9}\right) = \left(\frac{2}{3}\right)\left(\frac{2}{3}\right) = 1$. However, the equation $x^2 \equiv 2 \pmod 9$ has *no* solutions.

It seems that the primary importance of the Jacobi symbol is that it satisfies certain identities (including the law of quadratic reciprocity) that allow it (and consequently the Legendre symbol) to be calculated by a variant of Euclid's algorithm for the gcd. The following identities may be found in the texts of Niven and Zuckerman [12] and Vinogradov [15].

1. If $a \equiv b \pmod Q$ then $\left(\frac{a}{Q}\right) = \left(\frac{b}{Q}\right)$.
2. $\left(\frac{1}{Q}\right) = 1$.
3. $\left(\frac{-1}{Q}\right) = (-1)^{(Q-1)/2}$.
4. $\left(\frac{ab}{Q}\right) = \left(\frac{a}{Q}\right)\left(\frac{b}{Q}\right)$.
5. $\left(\frac{2}{Q}\right) = (-1)^{(p^2-1)/8}$.
6. If Q and P are relatively prime odd numbers then

   $\left(\frac{Q}{P}\right) = (-1)^{(P-1)(Q-1)/4}\left(\frac{P}{Q}\right)$. (Quadratic reciprocity)

Example 32: As an example of these identities, we compute $\left(\frac{117}{271}\right)$.

$$\begin{aligned}
\left(\frac{117}{271}\right) &= +\left(\frac{271}{117}\right) && \text{(by 6)}\\
&= +\left(\frac{37}{117}\right) && \text{(by 1)}\\
&= +\left(\frac{117}{37}\right) && \text{(by 6)}\\
&= +\left(\frac{6}{37}\right) && \text{(by 1)}\\
&= +\left(\frac{2}{37}\right)\left(\frac{3}{37}\right) && \text{(by 4)}\\
&= -\left(\frac{3}{37}\right) && \text{(by 5)}\\
&= -\left(\frac{37}{3}\right) && \text{(by 6)}\\
&= -\left(\frac{1}{3}\right) && \text{(by 1)}\\
&= -1 && \text{(by 2)}
\end{aligned}$$

Since 271 is a prime, we have in fact calculated the Legendre symbol $\left(\frac{117}{271}\right)$ and we may conclude that 117 has no square root modulo 271. Of course, another way to arrive at the same conclusion is to note that $117^{135} \equiv 270 \pmod{271}$.

Theorem 33: There is a polynomial time algorithm to compute the Jacobi symbol $\left(\frac{a}{Q}\right)$ whenever Q is an odd number greater than 1 and a is relatively prime to Q.

Proof: Devise an appropriate algorithm using the identities -- note that it is basically the same as Euclid's algorithm for the greatest common divisor, with some extra bookkeeping and special rules to cast out 2's. □

14. Recognizing primes: Solovay and Strassen's randomized procedure

This section describes Solovay and Strassen's Monte Carlo algorithm for testing primality [14].

Solovay and Strassen's algorithm

The input is an odd integer $n > 1$.

```
1. Choose at random $a \in \{1,2,...,n-1\}$.
2. If $\gcd(a,n) > 1$ then output "composite" and halt.
3. Calculate the quantities:
        $\delta = a^{(n-1)/2} \pmod{n}$,
        $\epsilon = \left(\frac{a}{n}\right)$  (The Jacobi symbol).
   a. If $\delta \not\equiv \epsilon$ (mod n) then output "composite" and halt.
   b. If $\delta \equiv \epsilon$ (mod n) then output "possibly prime" and halt.
```

Theorem 34:

1. This randomized procedure runs in polynomial time.
2. If n is prime then this procedure must output "possibly prime"
3. If n is composite then this procedure outputs "composite" with probability at least 1/2.

Proof: For (1). we have seen in previous sections that gcd, exponentiation modulo n, and the Jacobi symbol can be computed in polynomial time.

For (2), suppose that n is prime. Then $\gcd(a,n) = 1$ and we have seen that

$$a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n},$$

so neither step (2) nor step (3a) can output "composite" for n. Hence the output must be "possibly prime".

For (3), suppose that n is composite. We must argue that for at least half the possible choices of a, the output will be "composite". Define

$$S = \{a \in \mathbf{Z}_n^* : a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n}\}.$$

The elements of S are the only choices of a which will lead to an output of "possibly prime". We show that they comprise at most half of $\{1,2,\ldots,n-1\}$.

S is a subgroup of $\mathbf{Z}_n^*$ because it is closed under product (using the identity $\left(\frac{ab}{n}\right) = \left(\frac{a}{n}\right)\left(\frac{b}{n}\right)$ for the Jacobi symbol). Thus $|S|$ must divide $|\mathbf{Z}_n^*|$, so either

$$S = \mathbf{Z}_n^*,$$

or

$$|S| \leq (1/2)|\mathbf{Z}_n^*| \leq (n-1)/2.$$

We now show that the first alternative is impossible.

Assume that

$$a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n} \text{ for all } a \in \mathbf{Z}_n^*.$$

Then also

$$a^{n-1} \equiv 1 \pmod{n} \text{ for all } a \in \mathbf{Z}_n^*.$$

Let $n = p_1^{\alpha_1}p_2^{\alpha_2}\cdots p_k^{\alpha_k}$ be the prime factorization of n. We consider two cases: either $\alpha_i \geq 2$ for some i, or $\alpha_i = 1$ for all $i = 1,2,\ldots,k$.

Suppose i is such that $\alpha_i \geq 2$. Let $m = p_i^{\alpha_i}$. $\mathbf{Z}_m^*$ is a cyclic group of order $\phi(p_i^{\alpha_i}) = (p_i - 1)p_i^{\alpha_i-1}$. Choose a generator g of $\mathbf{Z}_m^*$. By the Chinese Remainder Theorem, we may choose a to be the unique element of $\mathbf{Z}_n$ such that

$$a \equiv g \pmod{m} \text{ and } a \equiv 1 \pmod{n/m}.$$

Then a must be relatively prime to n, i.e., $a \in \mathbf{Z}_n^*$. By our assumptions then

$$a^{n-1} \equiv 1 \pmod{n}, \text{ and}$$

$$g^{n-1} \equiv 1 \pmod{m}.$$

Since the order of g is $p_i^{\alpha_i - 1}(p_i - 1)$, we must have

$$p_i^{\alpha_i - 1}(p_i - 1) \mid (n-1).$$

Since $\alpha_i > 1$, this implies that $p_i \mid (n-1)$. But $p_i \mid n$, so $p_i \mid 1$, a contradiction.

For the second case, assume that $\alpha_i = 1$ for all $i = 1,2,...,k$. Thus,

$$n = p_1 \cdot p_2 \cdots p_k.$$

Since n is composite, we must have $k \geq 2$. In this case we choose g to be a generator of $\mathbf{Z}_{p_1}^*$, and, using the Chinese Remainder Theorem, $a \in \mathbf{Z}_n^*$ such that

$$a \equiv g \pmod{p_1} \text{ and } a \equiv 1 \pmod{n/p_1}.$$

Then a is relatively prime to n, so by our assumptions,

$$a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n}.$$

But by the definition of the Jacobi symbol,

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)\left(\frac{a}{p_2}\right) \cdots \left(\frac{a}{p_k}\right).$$

However, $a \equiv g \pmod{p_1}$, so $\left(\frac{a}{p_1}\right) = \left(\frac{g}{p_1}\right)$. Also, since p_j divides n/p_1 for $j \neq 1$ and $a \equiv 1 \pmod{n/p_1}$, we get $a \equiv 1 \pmod{p_j}$ for $j \neq 1$. Hence,

$$\left(\frac{a}{n}\right) = \left(\frac{g}{p_1}\right)\left(\frac{1}{p_2}\right) \cdots \left(\frac{1}{p_k}\right).$$

Thus, $\left(\frac{a}{n}\right) = \left(\frac{g}{p_1}\right) = -1$ because p_1 is prime and g is a primitive root of p_1. Thus,

$$a^{(n-1)/2} \equiv -1 \pmod{n}.$$

Thus,

$$a^{(n-1)/2} \equiv -1 \pmod{n/p_1},$$

which contradicts the fact that

$$a \equiv 1 \pmod{n/p_1}$$

because $k \geq 2$ and n odd implies

$$1 \not\equiv -1 \pmod{n/p_1}.$$

Thus, in both cases the assumption that

$$a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n} \text{ for all } a \in \mathbf{Z}_n^*$$

leads to a contradiction, so we conclude that $S \neq \mathbf{Z}_n^*$. Hence, $|S| \leq (n-1)/2$, so fewer than half the choices of $a \in \{1,2,...,n-1\}$ lead to the output "possibly prime". □

(The proof above is more complex than that given in Solovay and Strassen's paper [14]; the latter is somewhat incorrect.)

Note that if the random choices of a are independent in successive runs of the algorithm, we improve the probability of error. In particular, if n is composite then

$$\text{Prob(} k \text{ successive runs output "possibly prime")} \leq (1/2)^k.$$

15. Carmichael's Theorem

Theorem 35: Let n be an odd number greater than 1. Then $\lambda(n) \mid (n-1)$ if and only if for all $a \in \mathbf{Z}_n^*$ we have $a^{n-1} \equiv 1 \pmod{n}$.

Proof: Suppose $n = {p_1}^{\alpha_1}{p_2}^{\alpha_2}\cdots{p_m}^{\alpha_m}$ is the prime factorization of n. Recalling the definition of Carmichael's lambda function,

$$\lambda(n) = \text{lcm}\ \{\phi({p_1}^{\alpha_1}), \ldots, \phi({p_m}^{\alpha_m})\}.$$

Suppose $\lambda(n) \mid (n-1)$, and $\gcd(a,n) = 1$. Fix i, $1 \leq i \leq m$. Then $\gcd(a,{p_i}^{\alpha_i}) = 1$, so by Euler's Theorem,

$$a^{\phi({p_i}^{\alpha_i})} \equiv 1 \pmod{{p_i}^{\alpha_i}}.$$

Since $\phi({p_i}^{\alpha_i}) \mid \lambda(n)$ and $\lambda(n) \mid (n-1)$, we have

$$a^{n-1} \equiv 1 \pmod{{p_i}^{\alpha_i}}.$$

This is true for each of the pairwise relatively prime moduli ${p_1}^{\alpha_1},{p_2}^{\alpha_2},\ldots,{p_m}^{\alpha_m}$, so we have by the Chinese Remainder Theorem,

$$a^{n-1} \equiv 1 \pmod{n}.$$

Conversely, suppose that $a^{n-1} \equiv 1 \pmod{n}$ for all a such that $\gcd(a,n) = 1$. Fix i, $1 \leq i \leq m$. $\mathbf{Z}^*_{{p_i}^{\alpha_i}}$ is a *cyclic* group of order $\phi({p_i}^{\alpha_i})$, so we may choose g to be an element of order $\phi({p_i}^{\alpha_i})$. By the Chinese Remainder Theorem, we may find an element $a \in \mathbf{Z}_n$ such that $a \equiv g \pmod{{p_i}^{\alpha_i}}$ and $a \equiv 1 \pmod{{p_j}^{\alpha_j}}$ for all $j = 1,2,\ldots,m$ such that $j \neq i$. Then $\gcd(a,n) = 1$ and by our hypothesis, $a^{n-1} \equiv 1 \pmod{n}$. Then

$$a^{n-1} \equiv 1 \pmod{{p_i}^{\alpha_i}},$$

so

$$g^{n-1} \equiv 1 \pmod{{p_i}^{\alpha_i}}.$$

But the order of g is $\phi(p_i^{\alpha_i})$, so we must have

$$\phi(p_i^{\alpha_i}) \mid (n-1).$$

This is true for each $i = 1,2,\ldots,m$, so $(n-1)$ is a common multiple of each of $\phi(p_1^{\alpha_1}),\ldots,\phi(p_m^{\alpha_m})$. But $\lambda(n)$ is the least common multiple of these numbers, so we conclude that $\lambda(n) \mid (n-1)$. □

Corollary 36: In particular, for the composite number 561,

$$a^{560} \equiv 1 \pmod{561}$$

for all a such that $\gcd(a,561) = 1$.

Proof: $561 = 3\cdot 11\cdot 17$, so $\lambda(561) = \text{lcm}\ \{2, 10, 16\} = 80$. Thus, $\lambda(561) \mid 560$. □

(Remark: numbers n such that $\lambda(n) \mid n-1$ are called Carmichael numbers. It is not known whether there are infinitely many Carmichael numbers.)

16. Square roots of 1

We have shown that if p is prime then the only square roots of 1 are 1 and -1. However, 1 has *four* square roots modulo 15, namely: 1, 4, -1, -4. We show that this is a general phenomenon.

Theorem 37: Let n be divisible by two distinct odd prime numbers. Then 1 has at least four distinct square roots modulo n.

Proof: Let p be an odd prime number dividing n. Let $\alpha \geq 1$ be the largest integer such that $p^\alpha \mid n$. Then $n = p^\alpha \cdot m$, where $\gcd(m,p^\alpha) = 1$ and $m > 2$.

By the Chinese Remainder Theorem there exist unique $a, b \in \mathbf{Z}_n$ such that

$a \equiv 1 \pmod{p^\alpha}$ and $a \equiv -1 \pmod{m}$,

$b \equiv -1 \pmod{p^\alpha}$ and $b \equiv 1 \pmod{m}$.

Since $1 \not\equiv -1$ modulo either m or p^α, we have that $1, -1, a, b$ are all distinct modulo n. We have:

$a^2 \equiv 1 \pmod{p^\alpha}$ and $a^2 \equiv 1 \pmod{m}$,

$b^2 \equiv 1 \pmod{p^\alpha}$ and $b^2 \equiv 1 \pmod{m}$.

So, by another application of the Chinese Remainder Theorem,

$a^2 \equiv b^2 \equiv 1 \pmod{n}$.

Hence $1, -1, a, b$ are distinct square roots of 1 modulo n. □

Miller's procedure for testing primality uses the detection of one of these "peculiar" square roots of 1 (i.e., not 1 or -1) as one of the key methods of discovering the compositeness of n.

17. Recognizing primes: Miller's procedure

The primality testing procedure of Miller [11], like that of Solovay and Strassen, is based on searching for a value $a \in \{1,2,...,n-1\}$ that is a "witness" to the compositeness of n. The difference is that deterministic search replaces random selection. Values $a = 2,3,5,7,...$ are tried until either one is found that is a witness to the compositeness of n, or some bound is reached (in which case n is declared to be prime). The problem of the existence of "small" witnesses a for a composite n is efficiently reduced to the existence of certain "small" nonresidues. Work of Ankeny shows that the Extended Riemann Hypothesis (abbreviated ERH in what follows) implies a bound of $O(\log n)$ on the least such nonresidues, which in turn implies a polynomial time algorithm to recognize primes. (The reader is referred to Miller's paper for further details of the work of Ankeny and the ERH.) We now describe Miller's procedure.

Miller's procedure for testing primality

The input is an odd number $n > 1$.

Let s be the largest integer such that $2^s \mid n-1$, and let $Q = (n-1)/2^s$.
Let K be a fixed constant and let $f(n) = \min\{\lceil K(\log n^2)\rceil, n\}$.

```
1. If $n$ is a perfect power (i.e., $n = m^k$ for some $k \geq 2$) then output "composite" and halt.

2. Let $a$ run through all primes less than $f(n)$ in:

    a. If $a \mid n$ then output "composite" and halt.

    b. If $a^{n-1} \not\equiv 1 \pmod{n}$ then output "composite" and halt.

    c. If $a^Q \not\equiv 1 \pmod{n}$ then set $J = \max\{j : a^{2^jQ} \not\equiv 1 \pmod{n}\}$
       and if $a^{2^JQ} \not\equiv -1 \pmod{n}$ then output "composite" and halt.

3. If all primes $a < f(n)$ have been tested in step (2) without halting then
```

output "prime" and halt.

Theorem 38: For any constant K, Miller's procedure runs in deterministic polynomial time.

Proof: Testing whether n is a perfect power can be done in polynomial time (the only possible values of k are $2,3,\ldots,\lfloor log\ n \rfloor$). Since $f(n) = O((log\ n)^2)$ all primes $a < f(n)$ may be found in polynomial time with a sieve. Checking whether a divides n and computing the appropriate powers of a can be done in polynomial time. □

The following are the required definitions and theorems on least nonresidues.

Definition 39: If p is a prime and q is a prime dividing $p - 1$ then let
$N(p,q) =$ the least $q-th$ nonresidue mod p.

That is, $N(p,q)$ is the least nonnegative integer a such that $x^q \equiv a \pmod{p}$ has no solution.

Theorem 40: (Ankeny) The ERH implies that $N(p,q) = O((log\ p)^2)$.

Definition 41: If p and q are distinct primes, define $N(pq)$ to be the least positive integer a such that $\left(\frac{a}{pq}\right) = -1$.

Theorem 42: (Ankeny) The ERH implies that $N(pq) = O((log\ pq)^2)$.

(Remark: by index arguments mod p and q it may be seen that $N(p,q)$ and $N(pq)$ are both necessarily prime.)

Theorem 43: The ERH implies that there exists a constant K such that Miller's procedure correctly recognizes the prime numbers.

Proof: We consider the two cases: n is prime, and n is composite.

Suppose n is prime. Then n is not a perfect power and for each prime $a < f(n)$, $a \nmid n$. Also $a^{n-1} \equiv 1 \pmod{n}$ by Fermat's Theorem, and since 1 can only have the square roots 1 and -1 modulo a prime, we must have $a^{2^J Q} \not\equiv 1 \pmod{n}$ and $a^{2^{J+1} Q} \equiv 1 \pmod{n}$ implies that $a^{2^J Q} \equiv -1 \pmod{n}$. Hence for each a tested, none of steps (2a), (2b), or (2c) can output "composite". Thus the procedure correctly outputs "prime", independent of K.

Suppose that n is composite. If n is a power of a single prime, then n is a perfect power, so step (1) will output "composite". Thus, we may assume that n is not a prime power. Let $n = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$ be the prime factorization of n. Recall the definitions of Carmichael's and Miller's lambda functions:

$$\lambda(n) = \text{lcm } \{\phi(p_1^{\alpha_1}), \ldots, \phi(p_k^{\alpha_k})\},$$
$$\lambda'(n) = \text{lcm } \{p_1 - 1, \ldots, p_k - 1\}.$$

Note that $\lambda'(n) \mid \lambda(n)$. We consider two cases based on whether $\lambda'(n)$ divides $n - 1$ or not.

Suppose $\lambda'(n) \nmid n - 1$. Then clearly $\lambda(n) \nmid n - 1$, so by Carmichael's Theorem, there exists $a \in \mathbf{Z}_n^*$ such that $a^{n-1} \not\equiv 1 \pmod{n}$. Furthermore, we have:

Proposition 44: There exist primes p and q and an integer m such that

1. $p \mid n$, $(p - 1) \nmid (n - 1)$, and $q^m \mid (p - 1)$, $q^m \nmid (n - 1)$.

2. If a is any $q-th$ nonresidue modulo p then $a^{n-1} \not\equiv 1 \pmod{n}$.

Thus, if $a = N(p,q)$ then $a = O((log\ p)^2) = O((log\ n)^2)$ by Ankeny's Theorem, and a would lead to an output of "composite" in step (2b).

For the other case, assume that $\lambda'(n) \mid (n-1)$. We distinguish two subcases within this one. Let $\lambda'(n) = 2^l Q_1$, where Q_1 is odd. (Note that $Q_1 \mid Q$ because $\lambda'(n) \mid (n-1)$ and $(n-1) = 2^s Q$.) There must be some prime divisor of n, say p, such that $2^l \mid (p-1)$. Clearly, $(p-1) = 2^l Q_2$ where Q_2 is odd. The two cases we consider are based on whether there exists a prime q dividing n such that 2^l does not divide $(q-1)$.

Suppose there exists a prime q such that $2^l \nmid (q-1)$. In this case we write $q - 1 = 2^m Q_3$, where Q_3 is odd and $m < l$. Choose $a = N(p,2)$. We now show that a would lead to an output of "composite" in step (2). Suppose $a \nmid n$ and $a^{n-1} \equiv 1 \pmod{n}$. Then since a is a quadratic nonresidue mod p (this is how it was chosen),

$$a^{(p-1)/2} \equiv \left(\frac{a}{p}\right) \equiv -1 \pmod{p}.$$

Since Q is odd,

$$(a^Q)^{(p-1)/2} \equiv (-1)^Q \equiv -1 \pmod{p}.$$

Thus, $a^Q \not\equiv 1 \pmod{n}$ and the body of step (2c) must be executed.

Let $r = l - 1$. Note that $2^r Q$ is a multiple of $q - 1$ because $m \leq r$. Also, $2^r Q$ is an odd multiple of $(p-1)/2$. Thus

$$a^{2^r Q} \equiv (a^{q-1})^u \equiv 1 \pmod{q},$$

$$a^{2^r Q} \equiv (a^{(p-1)/2})^{2t+1} \equiv (-1)^{2t+1} \equiv -1 \pmod{p}.$$

Suppose $a^{2^J Q} \equiv -1 \pmod{n}$. Then $a^{2^J Q} \equiv -1 \pmod{q}$, so $J < r$. But also $a^{2^J Q} \equiv -1 \pmod{p}$, so $J = r$, a contradiction. Thus $a^{2^J Q} \not\equiv -1 \pmod{n}$, and a leads to an output of "composite" in step (2c).

For the second subcase, assume that for all primes q dividing n, 2^l divides $(q-1)$. In this case, let q be any prime distinct from p which divides n. Then $q = 2^l Q_3$, where Q_3 is odd. Choose $a = N(pq)$. We show that this a would lead to an output of "composite" in step (2).

Suppose that $a \nmid n$ and $a^{n-1} \equiv 1 \pmod{n}$. Since

$$-1 = \left(\frac{a}{pq}\right) = \left(\frac{a}{p}\right)\left(\frac{a}{q}\right),$$

we may assume without loss of generality that $\left(\frac{a}{p}\right) = -1$ and $\left(\frac{a}{q}\right) = 1$. Then

$$(a^Q)^{(p-1)/2} \equiv (-1)^Q \equiv -1 \pmod{p}$$

so $a^Q \not\equiv 1 \pmod{n}$, and the body of step (2c) must be executed if this a is tested in step (2). Let $r = l - 1$. Then $2^r Q$ is an odd multiple of $(p-1)/2$ and of $(q-1)/2$. Thus

$$a^{2^r Q} \equiv (a^{(p-1)/2})^{2u+1} \equiv (-1)^{2u+1} \equiv -1 \pmod{p},$$
$$a^{2^r Q} \equiv (a^{(q-1)/2})^{2t+1} \equiv (1)^{2t+1} \equiv 1 \pmod{q}.$$

Let us assume that $a^{2^J Q} \equiv -1 \pmod{n}$. Then $a^{2^J Q} \equiv -1 \pmod{p}$, so $J = r$, and $a^{2^J Q} \equiv -1 \pmod{q}$, so $J < r$, which is a contradiction. Thus, $a^{2^J Q} \not\equiv -1 \pmod{n}$, and a leads to an output of "composite" in step (2c) if a is tested in step (2).

Thus in either of these two subcases, there exists an element $a = O((log\ n)^2)$ which must lead to an output of "composite" if it is tested.

Thus if K is chosen to exceed the constants implied in the applications of Ankeny's Theorems, then the ERH implies that Miller's procedure will output "composite" for all composite inputs n. □

To conclude this section, we give a proof of Proposition 44.

Proof: If $\lambda'(n) \nmid (n-1)$ then for some p such that $p \mid n$, $(p-1) \nmid (n-1)$. If every prime power that divides a number a also divides the number b then $a \mid b$, so there exists a prime q such that $q^m \mid (p-1)$ and $q^m \nmid (n-1)$ for some $m \geq 1$.

Let a be any q*—th* nonresidue mod p. Let g be a primitive root modulo p and suppose $a \equiv g^r \pmod{p}$. Assume $a^{n-1} \equiv 1 \pmod{n}$. Then

$$(g^r)^{n-1} \equiv 1 \pmod{p}.$$

Since the order of g is $p-1$, this implies that $(p-1) \mid r(n-1)$. Thus $q^m \mid r(n-1)$ while $q^m \nmid (n-1)$. Hence $q \mid r$, say $r = t{\cdot}q$. Then

$$a \equiv g^r \equiv (g^t)^q \pmod{p},$$

so a is a q*—th* residue modulo p, contradicting our choice of a. Thus, $a^{n-1} \not\equiv 1 \pmod{n}$. □

Restating the main result of this section: if the Extended Riemann Hypothesis is true, then there is a deterministic polynomial time algorithm to test primality.

18. Square roots of − 1

Let p be an odd prime number. Does -1 have a square root? Recall that

$$\left(\frac{-1}{p}\right) \equiv (-1)^{(p-1)/2} \pmod{p},$$

so

$$\left(\frac{-1}{p}\right) = \begin{array}{l} +1 \ \ if \ \ p \equiv 1 \pmod{4}, \\ -1 \ \ if \ \ p \equiv 3 \pmod{4}. \end{array}$$

That is, -1 has a square root mod p if and only if p is congruent to 1 modulo 4. Thus:

Theorem 45: If p is an odd prime and $a \in \mathbf{Z}_p^*$ then

1. If $p \equiv 3 \pmod{4}$ then one of a and $-a$ is a quadratic residue, the other is a quadratic nonresidue.

2. If $p \equiv 1 \pmod{4}$ then both a and $-a$ are quadratic residues or quadratic nonresidues.

Proof: $\left(\frac{-a}{p}\right) = \left(\frac{-1}{p}\right)\left(\frac{a}{p}\right)$. □

Thus, in the case $p \equiv 3 \pmod{4}$ we can easily lay hands on a quadratic nonresidue; for $p \equiv 1 \pmod{4}$ it appears to be not so easy.

In the case $p \equiv 1 \pmod{4}$, we may ask whether the square roots of -1 themselves have square roots. If $a^2 \equiv -1 \pmod{p}$ then we have

$$\left(\frac{a}{p}\right) \equiv a^{(p-1)/2} \pmod{p} \equiv \begin{array}{l} +1 \text{ if } p \equiv 1 \pmod{8}, \\ -1 \text{ if } p \equiv 5 \pmod{8}. \end{array}$$

Continuing in this way, if $p - 1 = 2^s Q$, where Q is odd, we can build a binary tree of height $s - 1$ whose root is -1 and such that the two sons of each node are its square roots modulo p, and the leaves are quadratic nonresidues. Two examples are shown in the following.

Example 46:

```
Tree of square roots of -1 for p = 41

                 40
                /  \
               9    32
              / \   / \
             3  38 14  27
```

```
Tree of square roots of -1 for p = 17

                 16
              /       \
           4             13
         /    \        /    \
        2     15      8      9
       / \    / \    / \    / \
      6  11  7  10  5  12  3  14
```

The trees illustrated in above have the interesting property that multiplying a leaf by some ancestor of the leaf produces another leaf in the tree whose location is predictable from the level of the ancestor, namely:

leaf × immediate ancestor → some leaf in other half of tree
leaf × grandfather → some leaf in other half of this half of tree

.

.

.

leaf × root → leaf in other half of this pair

For example, in the second tree, 6 is in the first eighth of the tree and its ancestors are 2, 4, 16. Mod 17, multiplying 6 by 2 gives 12, in the second half of the tree, by 4 gives 7, in the second quarter of the tree, by 16 gives 11, in the second eighth of the tree. This property is exploited in the procedure of Adleman, Manders, and Miller [3] to find square roots modulo p.

19. Finding square roots modulo n, overview

In the next few sections, the problem considered is that of solving the equation $x^2 \equiv a \pmod{n}$ given a and n. We first summarize the results to be described.

1. Determining whether $x^2 \equiv a \pmod{n}$ is solvable.
   a. If n is prime and $n \nmid a$, then calculating $\left(\frac{a}{n}\right)$ may be done in polynomial time and tells us whether the equation is solvable.
   b. If n is composite *and* we know the prime factorization of n, then Vinogradov gives easily computed necessary and sufficient conditions for the solvability of $x^2 \equiv a \pmod{n}$ when $\gcd(a,n) = 1$.
   (See [15], Chapter V, Section 4.)
2. Finding solutions of $x^2 \equiv a \pmod{n}$.
   a. If n is prime then there are randomized polynomial time (Las Vegas) procedures to find solutions of $x^2 \equiv a \pmod{n}$. Procedures of Berlekamp, and of Adleman, Manders, and Miller are described. The latter procedure can be modified to be a deterministic polynomial time procedure if the ERH is true.
   b. If n is composite *and* we have the prime factorization of n, then:
      i. The algorithm of Adleman, Manders, and Miller can be used to find *some* solution of $x^2 \equiv a \pmod{n}$.
      ii. Finding the *least* solution of $x^2 \equiv a \pmod{n}$ in positive integers is an NP-hard problem.

20. Finding square roots: Berlekamp's procedure

Solve: $x^2 \equiv a \pmod{p}$, where p is an odd prime and $\gcd(a,p) = 1$.

Berlekamp's procedure for finding square roots

1. Check that $\left(\frac{a}{p}\right) = 1$, i.e., that the equation is solvable.
2. Find a number γ such that $(\gamma^2 - a)$ is a quadratic nonresidue, i.e., $\left(\frac{\gamma^2-a}{p}\right) = -1$.
3. Compute $(x^{(p-1)/2} - 1) \bmod ((x - \gamma)^2 - a)$. The result will be a linear polynomial $\delta(x - \rho)$. Output $(\rho - \gamma)$ as a square root of a.

Theorem 47: If a is a quadratic residue mod p and Berlekamp's procedure terminates, then its output is a square root of a.

Proof: Suppose a is a quadratic residue and the procedure succeeds in finding a suitable γ. Since $x^2 - a \equiv 0 \pmod{p}$ is solvable, $(x - \gamma)^2 - a \equiv 0 \pmod{p}$ is also solvable. Suppose ρ and σ are the solutions of the latter equation. Then

$$\rho\sigma \equiv (\gamma^2 - a) \pmod{p},$$

so, since $\left(\frac{\rho\sigma}{p}\right) = \left(\frac{\rho}{p}\right)\left(\frac{\sigma}{p}\right)$ and $\left(\frac{\gamma^2 - a}{p}\right) = -1$, we must have

$$\left(\frac{\rho}{p}\right) = -\left(\frac{\sigma}{p}\right).$$

Without loss of generality, assume that $\left(\frac{\rho}{p}\right) = 1$. Then $(x - \rho)$ is a factor of $x^{(p-1)/2} - 1$ modulo p, but $(x - \sigma)$ is not. Thus the greatest common divisor of $(x - \gamma)^2 - a$ and $x^{(p-1)/2} - 1$ is $(x - \rho)$. Hence computing

$$(x^{(p-1)/2} - 1) \bmod ((x - \gamma)^2 - a)$$

will produce $\delta(x - \rho)$ and the value $(\rho - \gamma)$ is a square root of a. □

For the question of "finding" γ, note that if $p \equiv 3 \pmod 4$, then $\left(\frac{-a}{p}\right) = -\left(\frac{a}{p}\right) = -1$, so the choice $\gamma = 0$ suffices in this case. In the case $p \equiv 1 \pmod 4$, choose γ's at random from $\mathbf{Z}_p^*$ until one is found that has the desired property. For this case, we show that at least half the possible choices of γ have the desired property. Thus, this is an algorithm whose output is correct, but whose running time depends on random choices -- we show that the expected value of the running time is polynomial in the length of the input.

Simplified Cyclotomy Theory

Lemma 48: Let p be a prime such that $p \equiv 1 \pmod 4$ and let g be any primitive root of p. Define for each i and $j \in \{0,1\}$:

$$S_{ij} = \{\ (x,y) \in \mathbf{Z}_{p-1} \times \mathbf{Z}_{p-1} : x \equiv i \pmod 2,\ y \equiv j \pmod 2, \text{ and } g^x + 1 \equiv g^y \pmod p\ \}.$$

Then $|S_{00}| = -1 + (p-1)/4$.

Proof: Note that the sets S_{00}, S_{01}, S_{10}, S_{11} are pairwise disjoint. For each x except $x = (p-1)/2$, we have $g^x + 1 \not\equiv 0 \pmod p$ so there is a unique $y \in \mathbf{Z}_{p-1}$ such that $g^x + 1 \equiv g^y \pmod p$. Thus:

1. $|S_{00}| + |S_{01}| + |S_{10}| + |S_{11}| = p - 2$.

2. $|S_{11}| = |S_{10}|$. This is true because

$$g^{2m+1} + 1 \equiv g^{2n+1} \pmod p$$

implies

$$g^{-(2m+1)} + 1 \equiv g^{2(n-m)} \pmod p.$$

Thus, $(x,y) \mapsto (-x,\ y-x)$ is a one to one correspondence between S_{11} and S_{10}.

3. $|S_{10}| = |S_{01}|$. This is true because

$$g^{2m+1} + 1 \equiv g^{2n} \pmod{p}$$

implies

$$-g^{2n} + 1 \equiv -g^{2m+1} \pmod{p},$$

which in turn implies

$$g^{2n+(p-1)/2} + 1 \equiv g^{2m+1+(p-1)/2} \pmod{p},$$

and $(p-1)/2$ is even. Thus, $(x,y) \mapsto (y + (p-1)/2,\ x + (p-1)/2)$ is a one to one correspondence between S_{10} and S_{01}.

4. $|S_{11}| + |S_{10}| = (p-1)/2$. This is true because

$$S_{11} \cup S_{10} = \{\ (x,y) : x \equiv 1 \pmod{2} \text{ and } g^x + 1 \equiv g^y \pmod{p}\ \}.$$

Thus,

$$|S_{11}| + |S_{10}| = |S_{11} \cup S_{10}| = (p-1)/2.$$

So, combining (2), (3), and (4), we have

$$|S_{11}| = |S_{10}| = |S_{01}| = (p-1)/4,$$

and substituting this into (1), we obtain

$$|S_{00}| = -1 + (p-1)/4.$$

□

Lemma 49: Let p be prime. Suppose $p \equiv 1 \pmod{4}$. Let $a \in \mathbf{Z}_p^*$ be any element such that $\left(\frac{a}{p}\right) = 1$. Then at most half of the elements γ of $\mathbf{Z}_p^*$ have $\left(\frac{\gamma^2-a}{p}\right) = 1$.

Proof: Let g be any primitive root modulo p. Let S_{00} be defined with respect to g as in the preceding lemma. Define

$$R = \{\gamma \in \mathbf{Z}_p^* : \left(\frac{\gamma^2 - a}{p}\right) = 1\},$$

$$S = \{b \in \mathbf{Z}_p^* : \left(\frac{b-a}{p}\right) = 1 \text{ and } \left(\frac{b}{p}\right) = 1\}.$$

Then each element of S gives rise to two elements of R, so $|R| = 2|S|$.

Since $\left(\frac{a}{p}\right) = 1$ and $p \equiv 1 \pmod 4$, $\left(\frac{-a}{p}\right) = 1$, so $ind_g(-a)$ is some even integer, say $2m$, so $-a \equiv g^{2m} \pmod p$. Let $b \in S$. Then $ind_g(b)$ is even, say $2n$, and $ind_g(b-a)$ is even, say $2r$. So we have

$$g^{2n} + g^{2m} \equiv g^{2r} \pmod p,$$

so

$$g^{2(n-m)} + 1 \equiv g^{2(r-m)} \pmod p.$$

Thus, if v is $2(n-m)$ reduced modulo $p-1$ and w is $2(r-m)$ reduced modulo $p-1$, then $(v,w) \in S_{00}$.

Clearly, $b \mapsto (v,w)$ is a one to one map of S into S_{00}, so

$$|S| \leq |S_{00}| = -1 + (p-1)/4,$$

(by the preceding lemma), and

$$|R| = 2|S| \leq -2 + (p-1)/2.$$

□

(Remark: the interested reader may find this treated more generally under "cyclotomy" in Hall's *Combinatorial Theory*, p. 147.)

Theorem 50: With random selection of γ from $\mathbf{Z}_p^*$ until a γ is found such that $\left(\frac{\gamma^2-a}{p}\right) = -1$, the expected running time of Berlekamp's procedure is polynomial in the lengths of a and p.

Proof: Step (1) is the computation of the Legendre symbol, which can be done in time polynomial in the lengths of a and p. As each random choice in step (2) has a probability of at least 1/2 of succeeding, the expected number of choices required is bounded by 2. For each choice, we may compute $\left(\frac{\gamma^2-a}{p}\right)$ in polynomial time. In step (3) the computation of

$$(x^{(p-1)/2}-1) \ \bmod ((x-\gamma)^2-a)$$

can be done by successively squaring x and reducing it modulo $((x-\gamma)^2-a)$, as in the computation of $a^m \pmod{n}$. □

(Remark: The procedure described above is a special case of a part of the general procedure in Berlekamp's paper [5], to which the reader is referred for more details.)

21. Finding square roots: the procedure of Adleman, Manders, and Miller

Solve: $x^2 \equiv a \pmod{p}$, where p is an odd prime and $\gcd(a,p) = 1$.

AMM's procedure for finding square roots

1. Check that $x^2 \equiv a \pmod{p}$ is solvable, i.e., that $\left(\frac{a}{p}\right) = 1$.
2. Find the largest positive integer k such that $2^k \mid (p-1)$ and let $Q = (p-1)/2^k$.
3. If $a^Q \equiv 1 \pmod{p}$ then output $a^{(Q+1)/2}$ reduced modulo p and halt.
4. Find integers s and t such that $s{\cdot}2^k + t{\cdot}Q = 1$.
5. Choose at random $\gamma \in \mathbf{Z}_p^*$ until $\left(\frac{\gamma}{p}\right) = -1$.
6. Set l to $\gamma^Q \pmod{p}$.
7. Find the least integer J such that

   $a^{2^J Q} \equiv l^{2^r} \pmod{p}$

   for some integer $r \in [J+1, k-1]$, and let m be the least such r for this J.

   a. If $J = 0$, then output $a^{2^{k-1}s} \cdot l^{2^{m-1}t} \pmod{p}$ and halt.

   b. If $J > 0$, then set l to $l^{1+2^{k-m}} \pmod{p}$ and go back to the beginning of step (7).

Theorem 51: If a is a quadratic residue mod p and the procedure of Adleman, Manders, and Miller terminates, then its output is a square root of a mod p.

Proof: If the output is

$$b \equiv a^{(Q+1)/2}$$

from step (3) then $a^Q \equiv 1 \pmod{p}$, so

$$b^2 \equiv a^{Q+1} \equiv a \pmod{p}.$$

If the output is

$$b \equiv a^{2^{k-1}s} \cdot l^{2^{m-1}t} \pmod{p}$$

from step (7a), then

$$b^2 \equiv a^{2^k s}(l^{2^m})^t \equiv a^{2^k s} \cdot a^{Qt} \equiv a \pmod{p}$$

because $l^{2^m} \equiv a^Q \pmod{p}$ and $s \cdot 2^k + t \cdot Q = 1$. □

Theorem 52: The running time of the procedure of Adleman, Manders, and Miller is bounded by a polynomial in the lengths of p and a, plus the time required to find γ in step (5).

Proof: Since $\gcd(Q,2^k) = 1$, s and t in step (4) can be computed using the gcd algorithm. By previous results, the computations in steps (1), (2), (3), (4), and (6) can be done in polynomial time. Now assume that the procedure reaches step (7), so that $a^Q \not\equiv 1 \pmod{p}$, and a γ has been found such that $\left(\frac{\gamma}{p}\right) = -1$.

Inductively assume that the value of l is such that $l^{2^{k-1}} \equiv -1 \pmod{p}$. This is certainly true the first time step (7) is executed, since $l \equiv \gamma^Q \pmod{p}$, so that

$$l^{2^{k-1}} \equiv \gamma^{2^{k-1}Q} \equiv \gamma^{(p-1)/2} \equiv \left(\frac{\gamma}{p}\right) \equiv -1 \pmod{p}.$$

The first time step (7) is executed, since $a^Q \not\equiv 1 \pmod{p}$ and

$$a^{2^{k-1}Q} \equiv a^{(p-1)/2} \equiv \left(\frac{a}{p}\right) \equiv 1 \pmod{p},$$

we must have $k > 1$, and there exists a j, $0 \leq j < k-1$ such that

$$a^{2^j Q} \equiv -1 \equiv l^{2^{k-1}} \pmod{p}.$$

Hence some j with the required property exists and we may take J to be the least one. (J can be found in polynomial time.) Then $J < k-1$. Now suppose that $J > 0$ and m is the associated value in the range $[J+1,\ k-1]$. The new value of l will be $l' = l^{1+2^{k-m}}$.

For this value of l' we have

$$(l')^{2^{k-1}} \equiv l^{2^{k-1}} l^{2^{2k-m-1}},$$

so

$$(l')^{2^{k-1}} \equiv -1 \pmod{p},$$

because $m \leq k-1$. Also,

$$(l')^{2^{m-1}} \equiv l^{2^{m-1}} l^{2^{k-1}},$$

so

$$(l')^{2^{m-1}} \equiv -l^{2^{m-1}},$$

and finally

$$(l')^{2^{m-1}} \equiv a^{2^{J-1}Q},$$

because $a^{2^J Q} \equiv l^{2^m}$ but $a^{2^{J-1}Q} \not\equiv l^{2^{m-1}}$,so we must have $a^{2^{J-1}Q} \equiv -l^{2^{m-1}}$. Hence the inductive hypothesis is verified regarding l, and the new value of j cannot exceed $J-1$. Hence, step (7) is repeated at most $k-1$ times before J is reduced to 0 and the procedure terminates. Since $k = O(log\ p)$, we conclude that step (7) contributes only a polynomial amount of computation to the running time of this procedure. □

Theorem 53: If γ is selected at random in step (5) of the procedure of Adleman, Manders, and Miller, then the procedure runs in expected polynomial time in the lengths of p and a.

Proof: Above we saw that exactly half the elements of $\mathbf{Z}_p^*$ are quadratic nonresidues. Thus the chance of success at each choice of γ is 1/2, and the expected number of choices required is 2. The computation for each choice is done in polynomial time. □

We may replace the random choice of γ in step (5) by a deterministic search for the least element γ such that $\left(\frac{\gamma}{p}\right) = -1$. This does not affect the correctness of the procedure, and we obtain the following theorem.

Theorem 54: If the ERH is true, then the version of the procedure of Adleman, Manders, and Miller that searches in order $\gamma = 2,3,4,\ldots$, runs in deterministic polynomial time.

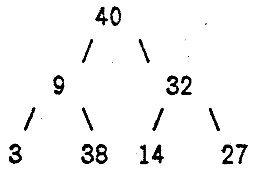
Proof: By Theorem 40, the ERH implies that the least quadratic nonresidue of p is of magnitude $O((\log p)^2)$. □

Remarks:

1. If $p \equiv 3 \pmod 4$, step 3 applies.
2. With respect to the tree of square roots of -1, a^Q is some element in the tree if $a^Q \not\equiv 1 \pmod p$, each value of l is a leaf of the tree, and replacing l by $l \cdot l^{2^{k-m}}$ corresponds to multiplying l by the appropriate ancestor to obtain a leaf in the "other" subtree of the common ancestor of a^Q and l in the tree. We thus narrow down to a leaf descended from a^Q.
3. For generalizations to roots other than the square root, see the paper of Adleman, Manders, and Miller [3].

Example 55: Solve $x^2 \equiv 21 \pmod{41}$.

The tree of square roots of -1 for p = 41:



Then we write $p - 1 = 40 = 2^2 \cdot 5$, so $k = 2$ and $Q = 5$.

Compute $21^5 \equiv 9$
$21^{10} \equiv 40$
$21^{20} \equiv 1$, so 21 has a square root mod 41.

Suppose we guess the quadratic nonresidue 6:

$6^5 \equiv 27$
$6^{10} \equiv 32$
$6^{20} \equiv 40$

Hence we have $27^2 \not\equiv 21^5$ and $27^4 \equiv 21^{10}$.

Compute $27^{1+2} \equiv 3$, giving a leaf in the other half of the tree from 27.

We find $3^2 \equiv 21^5$.

Taking $1 = 2 \cdot 2^3 - 3 \cdot 5$, we have

$$\begin{aligned} 21 &\equiv 21^{16}(21^5)^{-3} \\ &\equiv 21^{16}(3^2)^{-3} \\ &\equiv (21^8 3^{-3})^2 \\ &\equiv (12)^2 \pmod{41} \end{aligned}$$

22. Finding SOME square root, composite modulus

Solve: $x^2 \equiv a \pmod{m}$ given the prime factorization $m = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$, and a such that $\gcd(a,m) = 1$.

It suffices to find $x_1, x_2, \ldots, x_k$ such that

$$x_i^2 \equiv a \pmod{p_i^{\alpha_i}} \text{ for } i = 1,2,\ldots,k,$$

since by the Chinese Remainder Theorem, we may find $y \in \mathbf{Z}_m$ such that

$$y \equiv x_i \pmod{p_i^{\alpha_i}} \text{ for } i = 1,2,\ldots,k.$$

Then

$$y^2 \equiv x_i^2 \equiv a \pmod{p_i^{\alpha_i}} \text{ for } i = 1,2,\ldots,k,$$

so by another application of the Chinese Remainder Theorem,

$$y^2 \equiv a \pmod{m}.$$

If p is an odd prime, it suffices to find x_0 such that $x_0^2 \equiv a \pmod{p}$; we show how this can be built up to a solution of $x^2 \equiv a \pmod{p^\alpha}$ for $\alpha > 1$.

If x_0 is a solution of $x^2 \equiv a \pmod{p}$, then p divides $(a - x_0^2)$, so let q be the quotient of $(a - x_0^2)$ and p. $2x_0$ is relatively prime to p^2, so choose r such that $2x_0 r \equiv 1 \pmod{p^2}$. Then

$$2x_0 pqr = 2x_0 r(a - x_0^2) \equiv (a - x_0^2) \pmod{p^2}.$$

Thus

$$(x_0 + pqr)^2 \equiv x_0^2 + (a - x_0^2) + p^2q^2r^2 \pmod{p^2},$$

and so

$$(x_0 + pqr)^2 \equiv a \pmod{p^2}.$$

Hence $x_1 \equiv (x_0 + pqr) \pmod{p^2}$ is a solution of the equation $x^2 \equiv a \pmod{p^2}$. This step may be repeated with p^2 for p; p^3 for p^2, x_1 for x_0, and x_2 for x_1 to obtain a solution to $x^2 \equiv a \pmod{p^3}$, and so on, until a solution is obtained to $x^2 \equiv a \pmod{p^\alpha}$.

This procedure can be carried out in polynomial time, so the problem is reduced to one of finding square roots modulo p, for which the procedures of Berlekamp or Adleman, Manders, and Miller can be used. (For the solution of $x^2 \equiv a \pmod{2^m}$ see Vinogradov [15].)

Example 56: Solve $x^2 \equiv 7 \pmod{27}$.

1. $x^2 \equiv 7 \equiv 1 \pmod{3}$ has solution $x_0 = 1$

   $q = (7 - 1)/3 = 2$

   $2x_0 = 2$, so $r = 5$ since $2 \cdot 5 \equiv 1 \pmod{9}$

2. Thus $x_1 = 4 \equiv 1 + 3 \cdot 2 \cdot 5 \pmod{9}$ is a solution of $x^2 \equiv 7 \pmod{9}$.

   $q = (7 - 16)/9 = -1$

   $2x_1 = 8$, so $r = 17$ since $8 \cdot 17 \equiv 1 \pmod{27}$

3. Thus $x_2 = 13 \equiv 4 + 9 \cdot (-1) \cdot 17 \pmod{27}$ is a solution of $x^2 \equiv 7 \pmod{27}$

And, indeed, $13^2 = 169 \equiv 7 \pmod{27}$.

23. Finding the LEAST square root, composite modulus

This section describes a result of Manders and Adleman [10] showing the following decision problem is NP-complete.

> Given integers a, H, m, to determine whether there is an integer x satisfying:
>
> (I) 1. $0 \leq x \leq H$, and
>
> 2. $x^2 \equiv a \pmod{m}$.

Theorem 57: This problem is NP-complete, even if the prime factorization of m is given.

Proof: The problem is clearly in NP: we may guess a nonnegative integer $x \leq H$ and check that the second condition of (I) is satisfied.

To see that it is complete in NP, we reduce to it the problem of deciding whether a propositional formula in conjunctive normal form with three literals per clause is satisfiable. (The latter problem is called 3SAT.) Let ϕ be a propositional formula of the appropriate form with variables $X_0, X_1, \ldots, X_{l-1}$ and clauses $C_0, C_1, \ldots, C_{m-1}$. Define for all i and j such that $i \in [0,l-1]$ and $j \in [0,m-1]$:

$$\epsilon_{ij} = \begin{cases} +1 & \textit{if } X_i \textit{ occurs in } C_j \\ -1 & \textit{if } \sim X_i \textit{ occurs in } C_j \\ 0 & \textit{if neither occurs in } C_j \end{cases}$$

(We assume that any clauses containing both X_i and $\sim X_i$ have been eliminated.) Let $n = l + 2m$ and let $p_0, p_1, \ldots, p_{n-1}$ be the first n "sufficiently large" primes (that is, greater than the $n-th$ root of $4n16^m$). Define for each $i = 0,1,\ldots,n-1$:

$$m_i = p_i^{\,n}.$$

Let $Q = m_0 \cdot m_1 \cdots m_{n-1}$, and define for each $i = 0,1,\ldots,n-1$:

$$n_i = Q/m_i.$$

For $i = 0,1,\ldots,l-1$ let λ_i be the least positive integer such that

$$\lambda_i n_i \equiv \sum_{j=0,m-1} \epsilon_{ij} 16^j \pmod{16^m}, \text{ and}$$

$$\lambda_i n_i \not\equiv 0 \pmod{p_i}.$$

For $i = l,l+1,\ldots,l+2m-1$ choose λ_i to be the least positive integer such that

$$\lambda_i n_i \equiv 16^{\lfloor (i-l)/2 \rfloor} \pmod{16^m}, \text{ and}$$

$$\lambda_i n_i \not\equiv 0 \pmod{p_i}.$$

Finally, let

$$\tau \equiv \sum_{j=0,m-1} 16^j$$

$$H = \sum_{i=0,n-1} \lambda_i n_i$$

$$m = 2 \cdot 16^m \cdot Q$$

$$a = (2 \cdot 16^m + Q)^{-1}(Q \cdot \tau^2 + 2 \cdot 16^m \cdot H^2) \pmod{m}.$$

(The inverse in the last expression is modulo m. Note that τ is odd.)

Lemma 58: The conditions (I) are equivalent to

1. $0 \leq x \leq H$,

(II) 2. $x^2 \equiv \tau^2 \pmod{2 \cdot 16^m}$,

3. $x^2 \equiv H^2 \pmod{Q}$.

Proof: By the Chinese Remainder Theorem. □

Lemma 59: The conditions (II) are equivalent to

1. $0 \leq |x| \leq H$,

(III) 2. $x \equiv \tau \pmod{16^m}$,

3. $x^2 \equiv H^2 \pmod{Q}$.

Proof: Verify that for τ odd we have

$$(x - \tau)(x + \tau) \equiv 0 \pmod{2^{k+1}}$$

if and only if

$$(x - \tau) \equiv 0 \pmod{2^k} \text{ or } (x + \tau) \equiv 0 \pmod{2^k}.$$

□

Lemma 60: The solutions of

1. $0 \leq |x| \leq H$,

(IV)

2. $x^2 \equiv H^2 \pmod{Q}$.

are precisely $x = \sum_{i=0,n-1} \alpha_i \lambda_i n_i$, where each $\alpha_i \in \{1,-1\}$.

Proof: Because $n_i n_j \equiv 0 \pmod{Q}$ if $i \neq j$ we have

$$H^2 \equiv \sum_{i=0,n-1} \lambda_i^2 n_i^2 \pmod{Q}, \quad \text{and}$$
$$x^2 \equiv \sum_{i=0,n-1} \alpha_i^2 \lambda_i^2 n_i^2 \pmod{Q}.$$

Since each $\alpha_i \in \{1,-1\}$, each $\alpha_i^2 = 1$, so

$$x^2 - H^2 \equiv 0 \pmod{Q}.$$

Also, $|x| \leq H$, and so all the x's of the indicated form are solutions of (IV).

Conversely, if y is any solution of (IV) then $Q \mid (y - H)(y + H)$. For each prime p_i it cannot be the case that p_i divides both $(y - H)$ and $(y + H)$, for then $p_i \mid 2H$, so $p_i \mid H$. Since $p_i \mid n_j$ for $i \neq j$, this implies that $p_i \mid \lambda_i n_i$, contrary to our choice of λ_i. We now define

$$\alpha_i = \begin{array}{ll} +1 & if\ m_i \mid (y - H) \\ -1 & if\ m_i \mid (y + H), \end{array}$$

and

$$x = \sum_{i=0,n-1} \alpha_i \lambda_i n_i.$$

If $m_i \mid (y - H)$ then

$$y \equiv H \equiv \lambda_i n_i \equiv \alpha_i \lambda_i n_i \equiv x \pmod{m_i}.$$

If $m_i \mid (y + H)$ then

$$y \equiv -H \equiv -\lambda_i n_i \equiv \alpha_i \lambda_i n_i \equiv x \pmod{m_i},$$

by our choice of α_i. Hence for all $i = 0,1,\ldots,n - 1$,

$$x \equiv y \pmod{m_i},$$

so by the Chinese Remainder Theorem,

$$x \equiv y \pmod{Q}.$$

We note that $|x| \leq H$ by construction.

By our choice of the p_i's and the fact that $\lambda_i < 2 \cdot 16^m$, each term of H is bounded by $Q/2n$, so $2H < Q$. But $|y| \leq H$ and $|x| \leq H$ imply $|x - y| \leq 2H < Q$ and since $x \equiv y \pmod{Q}$, we must have $x = y$. Thus y is of the indicated form. □

Thus conditions (1) and (3) of set (II) guarantee a collection of independent binary choices (whether $\alpha_i = 1$ or -1) in the solution space. We now go on to show that condition (2), $x \equiv \tau \pmod{16^m}$, constrains these binary choices to correspond to a satisfying assignment of truth values for ϕ according to the following scheme:

$$V(X_i) = \begin{array}{ll} T & if\, \alpha_i = +1, \\ F & if\, \alpha_i = -1. \end{array}$$

The first l α_i's correspond to an assignment; the remaining $2m$ α_i's are available as "padding" -- two α_i's for each clause to bring its "value" up to a standard figure if the chosen assignment satisfies it. The condition on the *j–th* hexadecimal digit of x corresponds to the condition of the *j–th* clause of ϕ.

Lemma 61: ϕ is satisfiable if and only if there exists a solution of (III).

Proof: Suppose there is a solution of (III). Then by the preceding lemma, it must be of the form

$$x = \sum_{i=0,n-1} \alpha_i \lambda_i n_i,$$

where each $\alpha_i \in \{1,-1\}$. Define a truth value assignment by

$$V(X_i) = \begin{array}{ll} T & if\, \alpha_i = +1, \\ F & if\, \alpha_i = -1. \end{array}$$

We show now that V satisfies the formula ϕ. From condition (2) of (III), $x \equiv \tau\,(\text{mod } 16^m)$, so

$$\sum_{i=0,n-1} \alpha_i \lambda_i n_i - \sum_{j=0,m-1} 16^j \equiv 0 \ \ (\text{mod } 16^m),$$

therefore

$$\sum_{i=0,l-1} \alpha_i \,(\sum_{j=0,m-1} \epsilon_{ij} 16^j) + \sum_{j=0,m-1} (\alpha_{l+2j} + \alpha_{l+2j+1} - 1) 16^j \equiv 0 \ (\text{mod } 16),$$

thus

$$\sum_{j=0,m-1} ((\sum_{i=0,l-1} \alpha_i \epsilon_{ij}) + \alpha_{l+2j} + \alpha_{l+2j+1} - 1)\ 16^j \equiv 0 \ (\text{mod } 16^m).$$

Letting

$$R_j = (\sum_{i=0,l-1} \alpha_i \epsilon_{ij}) + \alpha_{l+2j} + \alpha_{l+2j+1} - 1,$$

we note that $|\epsilon_{ij}| = 1$ for exactly three values of i for each j (because each clause contains exactly three literals), and each α_i is either 1 or -1, so for each j, $R_j \in [-6,4]$. Hence

$$\sum_{j=0,m-1} R_j 16^j \equiv 0 \ \ (\text{mod } 16^m)$$

if and only if

$$R_j = 0 \ \text{ for } j = 0,1,\ldots,m-1.$$

The possible values of

$$\sum_{i=0,l-1} \alpha_i \epsilon_i$$

are 3, 1, −1, −3 corresponding respectively to 3, 2, 1, or no literals of C_j satisfied by the assignment V. The only possible values of

$$\alpha_{l+2j} + \alpha_{l+2j+1} - 1$$

are 1, −1, −3, so in order that $R_j = 0$, the assignment must satisfy some literal in C_j, so V satisfies C_j. Since C_j was an arbitrarily chosen clause, V satisfies the formula ϕ.

Conversely, suppose that ϕ is satisfiable, and let V be any satisfying assignment. For $i = 0,1,\ldots,l-1$ define

$$\alpha_i = \begin{cases} +1 & if\, V(X_i) = T \\ -1 & if\, V(X_i) = F. \end{cases}$$

Then for each $j = 0,1,\ldots,m-1$ we evaluate

$$\sum_{i=0,l-1} \alpha_i \epsilon_{ij}.$$

This must be 3, 1, or −1 because V satisfies some literal in C_j, so $\alpha_i \epsilon_{ij} = 1$ for some i. Then respectively choose $(\alpha_{l+2j}, \alpha_{l+2j+1})$ to be (−1,−1), (1,−1), or (1,1) in order that

$$R_j = (\sum_{i=0,l-1} \alpha_i \epsilon_i) + \alpha_{l+2j} + \alpha_{l+2j+1} - 1 \;=\; 0.$$

Then we have

$$x = \sum_{i=0,n-1} \alpha_i \lambda_i n_i$$

satisfies conditions (1) and (3) of (III) by the preceding lemma, and

$$x - \tau \equiv \sum_{j=0,m-1} R_j 16^j \pmod{16^m},$$

from above,

$$x - \tau \equiv 0 \pmod{16^m}.$$

Thus x satisfies condition (2) and therefore is a solution of (III). □

To conclude the proof of the theorem, it is not difficult to verify that a, H, m may be constructed in polynomial time from the formula ϕ -- the primes are small (they may be taken to be the first n primes exceeding 16), and the rest depends on computing gcd, inverses, and the Chinese Remainder Theorem. It is also clear that m is such that its factorization may be given (or found in polynomial time) without affecting the result. □

24. Acknowledgement and Warning

I am happy to acknowledge my indebtedness to Gary Miller for many hours of enlightening conversation on these topics when we both were graduate students at Berkeley, and after. Manuel Blum's enthusiastic and repeated insistence is the reason I have undertaken to make these notes a more accessible than they were previously. Responsibility for the (inevitable!) errors is mine alone, however.

These notes were originally prepared as part of the postgraduate course in Artificial Intelligence and Computer Science at Edinburgh University in 1977-78, and are currently rather out of date. They contain no treatment of factorization, and no account of the work of Adleman, Rumely, and Pomerance on testing primality, nor of Adleman's work on the discrete logarithm problem. There are doubtless many other topics that should be included for a complete treatment, and many places where the current treatment is inadequate. I would be happy to receive any suggestions or comments on the notes.

I. Appendix: three tables of powers

MOD 11

	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1	1
2	2	4	8	5	10	9	7	3	6	1
3	3	9	5	4	1	3	9	5	4	1
4	4	5	9	3	1	4	5	9	3	1
5	5	3	4	9	1	5	3	4	9	1
6	6	3	7	9	10	5	8	4	2	1
7	7	5	2	3	10	4	6	9	8	1
8	8	9	6	4	10	3	2	5	7	1
9	9	4	3	5	1	9	4	3	5	1
10	10	1	10	1	10	1	10	1	10	1

MOD 15

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	4	8	1	2	4	8	1	2	4	8	1	2	4
4	4	1	4	1	4	1	4	1	4	1	4	1	4	1
7	7	4	13	1	7	4	13	1	7	4	13	1	7	4
8	8	4	2	1	8	4	2	1	8	4	2	1	8	4
11	11	1	11	1	11	1	11	1	11	1	11	1	11	1
13	13	4	7	1	13	4	7	1	13	4	7	1	13	4
14	14	1	14	1	14	1	14	1	14	1	14	1	14	1

MOD 9

	1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	1	1
2	2	4	8	7	5	1	2	4
4	4	7	1	4	7	1	4	7
5	5	7	8	4	2	1	5	7
7	7	4	1	7	4	1	7	4
8	8	1	8	1	8	1	8	1

References

[1] L. Adleman and K. Manders.
Reducibility, randomness, and intractability.
In *9th ACM Symposium on Theory of Computing*, pages 151-163. ACM, 1977.

[2] L. Adleman and K. Manders.
Reductions that lie.
In *20th IEEE Symposium on Foundations of Computer Science*, pages 397-410. IEEE, 1979.

[3] L. Adleman, K. Manders, and G. Miller.
On taking roots in finite fields.
In *18th IEEE Symposium on Foundations of Computer Science*, pages 175-177. IEEE, 1977.

[4] A. V. Aho, J. E. Hopcropft, and J. D. Ullman.
The Design and Analysis of Computer Algorithms.
Addison-Wesley, Reading, Mass., 1974.

[5] E. R. Berlekamp.
Factoring polynomials over large finite fields.
Mathematics of Computation 24:713-735, 1970.

[6] R. D. Carmichael.
On composite numbers which satisfy the Fermat congruence.
American Mathematical Monthly 19:22-27, 1912.

[7] M. R. Garey and D. S. Johnson.
Computers and Intractability: a Guide to the Theory of NP-Completeness.
Freeman, 1979.

[8] J. Gill.
Computational complexity of probabilistic Turing machines.
SIAM J. Comput. 6:675-695, 1977.

[9] D. E. Knuth.
The Art of Computer Programming, Vol. II: Seminumerical Algorithms.
Addison-Wesley, Reading, Mass., 1969.

[10] K. Manders and L. Adleman.
NP-complete decision problems for quadratic polynomials.
In *8th Annual ACM Symposium on Theory of Computing*, pages 23-29. ACM, 1976.

[11] G. Miller.
Riemann's hypothesis and tests for primality.
J. Comp. Sys. Sci. 13:300-317, 1976.

[12] I. Niven and H. S. Zuckerman.
An Introduction to the Theory of Numbers.
John Wiley and Sons, N. Y., 1960.

[13] V. Pratt.
Every prime has a succinct certificate.
SIAM J. Comput. 4:214-220, 1975.

[14] R. Solovay and V. Strassen.
A fast Monte-Carlo test for primality.
SIAM J. Comput. 6:84-85, 1977.

[15] I. M. Vinogradov.
Elements of Number Theory.
Dover, 1954.