

12 Primality proving

In this lecture, we consider the following problem: given a positive integer N , how can we efficiently determine whether N is prime or not? This question is intimately related to the problem of factoring N ; without a method for determining primality, we have no way of knowing when we have completely factored N . This is an important issue for probabilistic factorization algorithms such as ECM: if we attempt to factor a prime number N with ECM, the algorithm will never terminate. This problem is not unique to ECM; currently every known factorization algorithm that achieves a subexponential running time (even heuristically) is a randomized algorithm; in the absence of an explicit primality test most of these algorithms will simply fail to terminate on prime inputs.

Even if we are able to guarantee termination, there is still the issue of correctness. If a Monte Carlo algorithm outputs the factorization $N = pq$, it is easy to check whether the product of p and q is in fact equal to N . But how do we know that this is the *complete* factorization of N ? We need a way to unequivocally *prove* that p and q are both prime.

12.1 Classical primality tests

The most elementary approach to the problem is trial division: attempt to divide N by every integer $p \leq \sqrt{N}$. If no such p divides N , then N must be prime. This takes time $O(\sqrt{N}M(\log N))$, which is exponential in $\log N$.

Remark 12.1. This complexity bound can be slightly improved. Using fast sieving techniques [8, Alg. 3.2.2], we can enumerate the primes p up to \sqrt{N} in $O(\sqrt{N} \log N / \log \log N)$ time and then perform trial divisions by just the primes $p \leq \sqrt{N}$. Applying the prime number theorem and the Schönhage-Strassen bound, the sieving time dominates the cost of the divisions and the overall complexity of trial division becomes $O(\sqrt{N} \log N / \log \log N)$.

Many classical primality tests are based on Fermat's little theorem.

Theorem 12.2 (Fermat). *If N is prime, then for all $a \in \mathbb{Z}/N\mathbb{Z}$:*

$$a^N = a.$$

This implies that if $a^N \neq a$ for some $a \in \mathbb{Z}/N\mathbb{Z}$, then N cannot be prime. This gives us a way to efficiently prove that certain integers are composite. For example, $N = 91$ is not prime, since:

$$2^{91} \equiv 37 \pmod{91}.$$

But this does not always work. For example, $341 = 11 \cdot 31$ is not clearly not prime, but

$$2^{341} \equiv 2 \pmod{341}.$$

In this case, using a different value of a will work:

$$3^{341} \equiv 168 \pmod{341}$$

proves that 341 is not prime.

However, for certain composite integers N there is *no* choice of a that will work. Thus even if $a^N \equiv a \pmod{N}$ for every integer a , we cannot be sure that N is prime.

Definition 12.3. A *Carmichael number* is a composite integer N such that $a^N \equiv a \pmod N$ for every integer a .

The first four Carmichael numbers are 561, 1105, 1729, and 2821; for more, see sequence [A002997](#) in the On-Line Encyclopedia of Integer Sequences (OEIS). The largest known Carmichael number has about 300 billion decimal digits and more than 10 billion distinct prime factors [5]. The question of whether or not there are infinitely many Carmichael numbers was open for more than 80 years and finally settled in 1994.

Theorem 12.4 (Alford-Granville-Pomerance). *The set of Carmichael numbers is infinite.*

Proof. See [6]. □

The infinitude of Carmichael numbers implies that any approach based on Fermat's little theorem is doomed to fail for an infinite set of integers. We would like a criterion that is true if, *and only if*, N is prime. One candidate is the following theorem, which uses the Euler function $\phi(N) = |\mathbb{Z}/N\mathbb{Z}^*|$.

Theorem 12.5. *A positive integer N is prime if and only if $\phi(N) = N - 1$.*

Proof. The forward implication is clear: if N is prime, then $\phi(N) = N - 1$. For the converse, we note that $\phi(1) = 1 \neq 1 - 1$ and 1 is not prime, and if $N = p^k$, for some prime p and integer $k > 1$, then we have

$$\phi(N) = \phi(p^k) = p^k - p^{k-1} < p^k - 1 = N - 1.$$

Otherwise, we may write $N = ab$ with $a \perp b$ and the multiplicativity of ϕ implies

$$\phi(N) = \phi(ab) = \phi(a)\phi(b) \leq (a-1)(b-1) < ab - 1 = N - 1. \quad \square$$

One approach suggested by this theorem is to simply compute $\phi(N)$ and see whether it is equal to $N - 1$. However, computing $\phi(N)$ is very difficult, in general.¹ Fortunately, we can use Theorem 12.5 in a less obvious way, via the following lemma. We restrict our attention to odd integers, since it is easy to tell whether an even integer is prime or not.

Lemma 12.6. *Let $p = 2^s t + 1$ be prime, with t odd, and let a be an integer that is nonzero modulo p . Exactly one of the following holds:*

- (i) $a^t \equiv 1 \pmod p$, or
- (ii) $a^{2^i t} \equiv -1 \pmod p$, for some $0 \leq i < s$.

Proof. The map $\varphi: x \mapsto x^t$ is an endomorphism of the cyclic group $(\mathbb{Z}/p\mathbb{Z})^\times$ of order $2^s t$. The kernel of φ is cyclic of order t , and its image is thus cyclic of order 2^s . For any $a \in \mathbb{Z}/p\mathbb{Z}^*$, either $a \in \ker \varphi$, in which case (i) holds, or $\varphi(a)$ has order 2^k for some $0 < k < s$. In the latter case, for $i = k - 1$ the integer $a^{2^i t}$ has order 2 in $(\mathbb{Z}/p\mathbb{Z})^\times$ and must be equal to -1 , which is the unique element of order 2, and then (ii) holds. □

Definition 12.7. Let $N = 2^s t + 1$ be an odd integer, with t odd. An integer $a \not\equiv 0 \pmod N$ is a *witness* for (the compositeness of) N if both of the following hold:

- (i) $a^t \not\equiv 1 \pmod N$
- (ii) $a^{2^i t} \not\equiv -1 \pmod N$ for $0 \leq i < s$.

¹If N is the product of two primes, it is easy to show that computing $\phi(N)$ is as hard as factoring N , and under the Extended Riemann Hypothesis, this is true in general [12].

If a is a witness for an integer N , then N must be composite, by Lemma 12.6, and we say that a *witnesses* the compositeness of N . Prime numbers clearly have no witnesses. It is not immediately clear that every composite integer N necessarily has any witnesses, but this is true. In fact, if we pick a at random it is quite likely to be a witness, as independently proved by Monier [13] and Rabin [16].

Theorem 12.8 (Monier–Rabin). *Let N be an odd composite integer. The probability that a random integer $a \in [1, N - 1]$ is a witness for N is at least $3/4$.*

The theorem suggests that if N is composite and we pick, say, 100 random integers $a \in [1, N - 1]$, then we are almost certainly going to find a witness for N . On the other hand, if N is prime then we will not find a witness. This doesn't actually *prove* that N is prime (unless we try more than $1/4$ of all $a \in [1, N - 1]$), but we can at least view it as strongly supporting this possibility.

*Proof.*² Let N be an odd composite number of the form $N = 2^s t + 1$ with t odd, and let $N = q_1 \cdots q_r$ be the factorization of N into powers of distinct primes. Let us consider a particular $a \in [1, N - 1]$, and let $b = a^t$. If a is not a witness then either $b \equiv 1 \pmod N$, in which case $b \equiv 1 \pmod{q_j}$ for all of the q_j , or $b^{2^i} \equiv -1 \pmod N$ for some $0 \leq i < s$, in which case $b^{2^i} \equiv -1 \pmod{q_j}$ for all of the q_j . In either case, $b \pmod{q_j}$ is an element of the 2-Sylow subgroup of $(\mathbb{Z}/q_j\mathbb{Z})^\times$ for all q_j , and it has order 2^{i+1} in $(\mathbb{Z}/q_j\mathbb{Z})^\times$ for each q_j (with $i = -1$ in the first case). We will show that the probability of this happening for a random choice of a is at most $1/4$. We consider three cases.

Case 1: N is not square-free. Then some $q_j = p^k$ with $k > 1$. Since p is odd, the group $(\mathbb{Z}/p^k\mathbb{Z})^\times$ is cyclic of order $\phi(p^k) = p^{k-1}(p-1)$. The prime p cannot divide t , since t divides $N-1$ and p divides N . It follows that if a has order divisible by p in $(\mathbb{Z}/p^k\mathbb{Z})^\times$, then so does $b = a^t$, and in this case b cannot be an element of the 2-Sylow subgroup of $(\mathbb{Z}/p^k\mathbb{Z})^\times$. Thus the probability that b lies in the 2-Sylow subgroup of $(\mathbb{Z}/p^k\mathbb{Z})^\times$ is at most $1/p^{k-1}$, which is less than $1/4$ for all $p^k > 9$. For $p^k = 9$ one checks that for $t = \pm \pmod 6$, at most $2/9 < 1/4$ of the possible values of $a \pmod 9$ yield $b = a^t = \pm 1 \pmod 9$ in the 2-Sylow subgroup of $(\mathbb{Z}/9\mathbb{Z})^\times$.

Case 2: $r \geq 3$. For each q_j the 2-Sylow subgroup G_j of $(\mathbb{Z}/q_j\mathbb{Z})^\times$ is a cyclic of order 2^{k_j} , for some $k_j > 1$, and at most half the elements in G_j have any particular order. Assuming $b = a^t \pmod{q_j}$ lies in G_j for $j = 1, 2, 3$, the probability that it has the same order in each case is at most $1/4$.³

Case 3: $N = pq$ for distinct primes p and q . We may write $p = 2^{s_p} t_p + 1$, and $q = 2^{s_q} t_q + 1$, with t_p and t_q odd. Define the random variable X_p to be -1 if $b \pmod p$ does not lie in the 2-Sylow subgroup G_p of $(\mathbb{Z}/p\mathbb{Z})^\times$, and otherwise let $X_p = i$, where $b \pmod p$ has order 2^i in G_p . Similarly define the random variable X_q . We wish to show that $\Pr[X_p = X_q \geq 0] \leq 1/4$.

We first suppose $s_p > s_q$. Half the elements in the 2-Sylow subgroup of $(\mathbb{Z}/p\mathbb{Z})^\times$ have order $2^{s_p} > 2^{s_q}$, so $\Pr[0 \leq X_p \leq s_q] \leq 1/2$. We also have $\Pr[X_q = X_p | 0 \leq X_p \leq s_q] \leq 1/2$, thus $\Pr[X_p = X_q \geq 0] \leq 1/4$.

We now suppose that $s_p = s_q$. We have

$$2^s t = N - 1 = pq - 1 = (p-1)(q-1) + (p-1) + (q-1) = 2^s t_p t_q + 2^{s_p} t_p + 2^{s_p} t_q,$$

so if t_p divides t then it divides t_q and conversely. It follows that t_p and t_q cannot both divide t (if this were true then we would have $t_p = t_q$ and $p = q$ but we assumed $p \neq q$). So

²The proof we give here is a bit different (and more elementary) than the proofs of Monier and Rabin.

³This rules out all Carmichael numbers, since they all have at least 3 distinct prime factors.

we may assume without loss of generality that t_p does not divide t . This means $t_p \neq 1$, so t_p is divisible by an odd prime $\ell \geq 3$ that does not divide t . It follows that $\Pr[X_p \geq 0] \leq 1/3$, and we also have $\Pr[X_q = X_p | X_p \geq 0] \leq 1/2$, hence $\Pr[X_p = X_q \geq 0] \leq 1/6 < 1/4$. \square

Theorem 12.8 yields the following probabilistic primality test, due to Gary Miller [12] and Michael Rabin [16]

Algorithm 12.9 (Miller-Rabin). Given an odd integer N :

1. Pick a random integer $a \in [1, N - 1]$.
2. Write $N = 2^2t + 1$, with t odd, and compute $b = a^t \bmod N$.
If $b \equiv \pm 1 \pmod N$, return **true** (a is not a witness, N could be prime).
3. For i from 1 to $s - 1$:
 - a. Set $b \leftarrow b^2 \bmod N$.
 - b. If $b \equiv -1 \pmod N$, return **true** (a is not a witness, N could be prime).
4. Return **false** (a is a witness, N is definitely not prime).

Example 12.10. For $N = 561$, $a = 2$: $561 = 2^4 \cdot 35 + 1$. Thus, we compute:

$$2^{35} \equiv 263 \pmod{561}$$

This is not $\pm 1 \pmod{561}$ so we continue:

$$236^2 \equiv 166 \pmod{561}$$

$$166^2 \equiv 67 \pmod{561}$$

$$67^2 \equiv 1 \pmod{561}$$

None of these values is congruent to -1 , so $a = 2$ is a witness for $N = 561$ and we return **false**, meaning that 561 is definitely not a prime. Note the contrast with the Fermat test, which effectively just checks whether the last value computed above is 1 and does not detect that 561 is composite.

The Miller-Rabin test as a Monte Carlo algorithm with 1-sided error. If N is prime the algorithm will always correctly output **true**, and if N is composite the algorithm will correctly output **false** with probability at least $3/4$. The running time of the algorithm is $O(nM(n))$, where $n = \log N$, the same as exponentiation in $\mathbb{Z}/N\mathbb{Z}$. This makes it extremely efficient, and it is the most widely used method for testing primality. In practical implementations, one performs several iterations of the Miller-Rabin test (choosing a new random integer a each time), and if they all return **true**, conclude that N is “probably prime”.

But we should be careful how we interpret this. For any particular integer N , either N is prime or it isn't; it makes no sense to say that N is prime with some probability. If N is a random integer distributed over some interval, then it does make sense to ask for the probability that N is prime, given that it passed a Miller-Rabin test. However, if N is selected from a large interval, say $[1, e^{1000}]$, then the probability that N is prime is quite small, approximately $1/1000$. In this situation, we need to be careful, since false positives are more likely than primes. It might appear to require several Miller-Rabin tests before we could say with better than 50% confidence that a large random integer N is prime. However, the Miller-Rabin test is actually much more effective than Theorem 12.8 suggests.

Theorem 12.11 (Damgård-Landrock-Pomerance). *Let N be a random odd integer in $[2^{k-1}, 2^k]$. Let a be a random integer in $[1, N - 1]$. Then, if a is not a witness for N , then*

$$\Pr[N \text{ is prime}] \geq 1 - k^2 \cdot 4^{2-\sqrt{k}}.$$

Proof. See [9, Thm. 2]. □

Example 12.12. For large N , Theorem 12.11 gives excellent bounds on the probability that a random integer N is prime, given that it passes a single Miller-Rabin test:

$$\text{If } k = 256, \text{ then } 1 - k^2 \cdot 4^{2-\sqrt{k}} \geq 1 - 2^{-12}.$$

$$\text{If } k = 1024, \text{ then } 1 - k^2 \cdot 4^{2-\sqrt{k}} \geq 1 - 2^{-40}.$$

Thus when k is large it only takes a few successful Miller-Rabin tests to become astronomically confident that a randomly chosen integer N is prime.

12.2 Elliptic Curve Primality Proving

We now consider a method to unequivocally *prove* whether a given integer N is prime or composite, using elliptic curves. Elliptic curve primality proving (ECP) is based on a fundamental theorem of Goldwasser and Kilian [10]. To simplify the statement of the theorem, we make the following definitions.

Definition 12.13. Let $P = (x : y : z)$ be a projective point on an elliptic curve E/\mathbb{Q} , where $x, y, z \in \mathbb{Z}$, and let N be a nonzero integer. If $z \equiv 0 \pmod{N}$ then the point P is said to be *zero mod N* ; otherwise, P is *nonzero mod N* . If $\gcd(z, N) = 1$ then the point P is said to be *strongly nonzero mod N* .

Note that if P is strongly nonzero mod N , then P is nonzero mod p for every prime $p|N$. When N is prime, the notions of nonzero and strongly nonzero coincide. We now state the theorem, using $\Delta(E)$ to denote the discriminant of E (so $\Delta(E) = -16(4A^3 + 27B^2)$ for $E: y^2 = x^3 + Ax + B$ in Weierstrass form).

Theorem 12.14 (Goldwasser-Kilian). *Let E/\mathbb{Q} be an elliptic curve, and let M and N be positive integers with $M > (N^{1/4} + 1)^2$ and N prime to $\Delta(E)$. Suppose there is a point $P \in E(\mathbb{Q})$ such that MP is zero mod N and $(M/\ell)P$ is strongly nonzero mod N for every prime $\ell|M$. Then N is prime.*

Proof. Suppose for the sake of contradiction that the hypothesis holds and N is composite. Then N has a prime divisor $p \leq \sqrt{N}$, and E has good reduction at p since N is prime to $\Delta(E)$. Let M_p be the order of the reduction of P on E modulo p . The point MP is zero mod N and therefore zero mod p , so $M_p|M$; and we must have $M_p = M$, since $(M/\ell)P$ is strongly nonzero mod N and therefore nonzero mod p , for every prime $\ell|M$. Thus P has order M on the reduction of E modulo p , and by the Hasse bound $M \leq (\sqrt{p} + 1)^2$. But we also have $M > (N^{1/4} + 1)^2 \geq (p^{1/2} + 1)^2$, which is our desired contradiction. □

In order to apply the theorem, we need to know the prime factors q of M . In particular, we need to be sure that these q are actually prime! To simplify matters, we restrict ourselves to the case that $M = q$ is prime, and introduce the notion of a *primality certificate*.

Definition 12.15. A *primality certificate* for p is a tuple of integers

$$(p, A, B, x_1, y_1, q),$$

where $P = (x_1 : y_1 : 1)$ is a point on the elliptic curve $E: y^2 = x^3 + Ax + B$ over \mathbb{Q} , the positive integer p is prime to $\Delta(E)$, and $q > (p^{1/4} + 1)^2$ is such that qP is zero mod p .

Note that $P = (x_1 : y_1 : 1)$ is always strongly nonzero mod p , since its z -coordinate is 1. Theorem 12.14 implies that if there exists a primality certificate (p, \dots, q) for $N = p$ in which $M = q$ is prime, then p is prime. Thus a primality certificate (p, \dots, q) reduces the question of p 's primality to the question of q 's primality. Using a chain of such certificates, we can reduce to a case in which q is so small that we are happy to test its primality via trial division. This leads to the following recursive algorithm.

Algorithm 12.16 (Goldwasser-Kilian). Given an odd integer p (a candidate prime), and a bound b , with $p > b > 5$, either construct a primality certificate (p, A, B, x_1, y_1, q) with $q \leq (\sqrt{p} + 1)^2/2$, if p is prime, or return **composite** if p is composite.

1. Pick random integers $A, x_0, y_0 \in [0, p - 1]$, and set $B = y_0^2 - x_0^3 - Ax_0$.
Repeat until $\gcd(4A^3 + 27B^2, p) = 1$, then define $E: y^2 = x^3 + Ax + B$.
2. Use Schoof's algorithm to compute the number of points m on the reduction of E modulo p , assuming that p is prime. If anything goes wrong, or if $m \notin \mathcal{H}(p)$, then return **composite**.
3. Write $m = cq$, where $l \nmid c$ for primes $l > b$ and $l \nmid q$ for primes $l \leq b$.
If $c = 1$ or $q \leq (p^{1/4} + 1)^2$, then go to step 1.
4. Perform a Miller-Rabin test on q . If it returns **false** then go to step 1.
5. Compute $P = (P_x : P_y : P_z) = c \cdot (x_0 : y_0 : 1)$ on E , working modulo p .
If $\gcd(P_z, p) \neq 1$, go to step 1, else set $x_1 = P_x/P_z \bmod p$ and $y_1 = P_y/P_z \bmod p$.
6. Compute $Q = (Q_x : Q_y : Q_z) = q \cdot (x_1 : y_1 : 1)$ on E , working modulo p .
If $Q_z \not\equiv 0 \pmod p$ then return **composite**.
7. If $q > b$, then recursively verify that q is prime using inputs q and b ; otherwise, verify that q is prime by trial division. If q is found to be composite, go to step 1.
8. Output the certificate $(p, A, \tilde{B}, x_1, y_1, q)$, where $\tilde{B} \equiv B \pmod p$ is chosen so that we have $y_1^2 = x_1^3 + Ax_1 + \tilde{B}$ (over \mathbb{Z} not just modulo p).

Note that step 4 is not strictly necessary, a composite q would eventually be detected in the recursive call, but it greatly reduces the probability that we will waste time in the recursive call, which speeds up the algorithm.

When the input to Algorithm 12.16 is prime, it will output a sequence of certificates, one for each recursive call, that reduce the question of p 's primality to that of a prime $q < b$ that has been proved prime via trial division. Taken together, the sequence of primality certificates constitute a *primality proof* for p . The complexity of this algorithm, and the complexity of verifying the primality proof it generates, are considered in the problem set, under the heuristic assumption that the integer M behaves like a random integer.

Without any heuristic assumptions, Goldwasser and Kilian proved that for almost all inputs p of a given size (all but a subexponentially small fraction), the expected running time of this algorithm is polynomial in $\log p$. Heuristically, this is believed to be true for all inputs, but we cannot prove this. Adleman and Huang later came up with a clever work-around to this problem that yielded an algorithm with a provably polynomial expected

running time for all inputs [4]. Their strategy is to “reduce” the problem of proving the primality of the given input p that of proving the primality of a *larger* prime $p' \approx p^2$. The prime p' is obtained in a random way, making it very likely that the Goldwasser-Kilian algorithm can prove its primality within a polynomial time bound (and if this does not happen we can always generate a different p' and try again). In practice the algorithm of Adleman and Huang is never used, since it is believed that it in fact it is always faster to use the original Goldwasser-Kilian algorithm, no matter what p is. But their result was theoretically significant, since it proved for the first time that primes could be recognized in polynomial time by a randomized algorithm.

Remark 12.17. In [4] Adleman and Huang obtain the prime p' as the order of a randomly chosen abelian variety J_C of dimension 2 that is associated to a genus 2 curve C over \mathbb{F}_p (assuming that p is prime). The abelian variety J_C is called the *Jacobian* of the curve C , and is analogous to the group of points on an elliptic curve (an abelian variety of dimension 1), except that when C has genus 2 the “points” on J_C actually correspond to pairs of points on the curve C . There is a generalization of Hasse’s theorem due to Weil that implies that the cardinality of $J_C(\mathbb{F}_p)$ is on the order of p^2 and lies within an interval of width $\approx 8p^{3/2}$. This interval is large enough (relative to p^2) that we can prove that it contains many primes, roughly as many as implied by the prime number theorem. Adleman and Huang show that for a random curve C , the cardinality of $J_C(\mathbb{F}_p)$ is reasonably likely to be any one of a large subset of these primes, yielding a prime p' that is very likely to be one that the Goldwasser-Kilian algorithm can certify in polynomial time. In order to make this all work, Adleman and Huang modify the Goldwasser-Kilian algorithm slightly to make the proportion of bad inputs even smaller, and they also use the fact that $\#J_C(\mathbb{F}_p)$ can be computed in polynomial time using an analog of Schoof’s algorithm.

In fact, the original algorithm of Goldwasser-Kilian is no longer used; there is a faster version of this algorithm due to Atkin and Morain that uses the CM method to construct an elliptic curve E modulo p with suitable order M (assuming that p is prime), eliminating the need to generate many random curves, and the need to use Schoof’s algorithm [3]. Like the Goldwasser-Kilian algorithm, this algorithm has not been proved to run in expected polynomial time, but in practice it is very fast. When combined with a further optimization due to Shallit [14], its expected running time is heuristically believed to be $\tilde{O}(n^4)$, where $n = \log p$. This makes it the current method of choice for general purpose primality proving. We will be able to examine the Atkin-Morain algorithm more closely after we have studied the theory of complex multiplication.

We should note that a deterministic polynomial-time algorithm for proving primality was later developed by Agrawal, Kayal, and Saxena [2]. This is an important theoretical result, but is not used in practice. The time bound proved in [2] is $\tilde{O}(n^{10.5})$; this can be improved to $\tilde{O}(n^6)$ (see [11]), but this is still much slower than the $\tilde{O}(n^4)$ heuristic complexity of (fast) ECPP. There is a randomized version of the AKS algorithm due to Bernstein [7] that runs in $\tilde{O}(n^4)$ time, but the constant factors appear to make it slower than ECPP, and it requires more memory. The certificates it produces also take longer to verify: $\tilde{O}(n^4)$ versus $\tilde{O}(n^3)$.

The current record for general purpose elliptic curve primality proving involves a 29,271 digit prime and was set in December 2014 (see [15] for an up-to-date list of ECPP records). We should note that there are much larger integers that have been proved prime (for example, the Mersenne prime $2^{57,885,161} - 1$), but these are all of a special form that permit much faster $\tilde{O}(n^2)$ -time algorithms to be applied. There are also specialized forms of elliptic curve

primality proving that run in $\tilde{O}(n^2)$ -time and these have been used to prove the primality of some very large integers for which no non-elliptic curve based method is applicable; the largest such prime has 419,110 digits [1].

References

- [1] A. Abatzoglou, A. Silverberg, A.V. Sutherland, and A. Wong, *A framework for deterministic primality proving using elliptic curves with complex multiplication*, Mathematics of Computation, to appear.
- [2] M. Agrawal, N. Kayal, N. Saxena, *Primes is in P*, Annals of Mathematics **160** (2004), 781–793.
- [3] A.O.L. Atkin and F. Morain, *Elliptic curves and primality proving*, Mathematics of Computation **61** (1993), 29–68.
- [4] L. Adleman and M-D. A. Huang, *Primality testing and abelian varieties over finite fields*, Lecture Notes in Mathematics **1512**, Springer, 1992.
- [5] W. Alford, J. Grantham, S. Hayman, and A. Shallue, *Constructing Carmichael numbers through improved subset-product algorithms*, Mathematics of Computation **83** (2014), 889–915.
- [6] W.R. Alford, A. Granville, and C. Pomerance, *There are infinitely many Carmichael numbers*, Annals of Mathematics **140** (1994), 703–722.
- [7] D.J. Bernstein, *Proving primality in essentially quartic random time*, Mathematics of Computation **76** (2007), 389–403.
- [8] R. Crandall and C. Pomerance, *Prime numbers: A computation perspective*, 2nd edition, Springer, 2005.
- [9] I. Damgård, P. Landrock, and C. Pomerance, *Average case error estimates for the strong probable prime test*, Mathematics of Computation **61** (1993), 177–194.
- [10] S. Goldwasser and J. Kilian, *Almost all primes can be quickly certified*, Proceedings of the Eighteenth ACM Symposium on the Theory of Computing (1986), 316–329.
- [11] H. Lenstra and C. Pomerance, *Primality testing with Gaussian periods*, preprint, 2011.
- [12] G. L. Miller, *Riemann’s hypothesis and tests for primality*, Journal of Computer and System Sciences **13** (1976), 300–317.
- [13] L. Monier, *Evaluation and comparison of two efficient probabilistic primality testing algorithms*, Theoretical Computer Science **12** (1980), 97–108.
- [14] F. Morain, *Implementing the asymptotically fast version of the elliptic curve primality proving algorithm*, Mathematics of Computation **76** (2007), 493–505.
- [15] C.K. Caldwell, *The top twenty elliptic curve primality proofs*, Prime Pages website, accessed 2015.
- [16] M.O. Rabin *Probabilistic algorithm for testing primality*, Journal of Number Theory **12** (1980), 128–138.

MIT OpenCourseWare
<http://ocw.mit.edu>

18.783 Elliptic Curves
Spring 2015

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.