CS60082/CS60094 Computational Number Theory, Spring 2009

End-semester examination: Solutions

[*This test is open-notes. Answer all questions. Be brief and precise.*]

1 Suppose we want to compute the smallest prime $p \geqslant n$, where n is a given positive integer. Assume that $n \gg 1$. The obvious strategy is to test the primality of $n, n+1, n+2, \ldots$ until a prime $n+k$ is found. During the search, it is natural to exclude the integers which are *obviously* not prime. For example, there is no need to check the primality of even integers, the multiples of 3, the multiples of 5, and so on. A sieve can be used to throw away multiples of small primes $p_1, p_2, \ldots, p_t$ and check the primality of only those integers of the form $n+i$ that do not have prime divisors $\leqslant p_t$. One takes t between 10 and 1000.

A prime p is called a *Sophie Germain prime* if $2p+1$ is also a prime. It is conjectured that there are infinitely many Sophie Germain primes. If p is a Sophie Germain prime, the prime $2p+1$ is called a *safe prime*. Safe primes are frequently used in cryptography.

In this exercise, you are asked to extend the above sieve for locating the smallest Sophie Germain prime $p \geqslant n$ for a given positive integer $n \gg 1$. Sieve over the interval $[n, n+M]$.

(a) Determine a value of M such that there is (at least) one Sophie Germain prime of the form $n+i$, $0 \leqslant i \leqslant M$, with high probability. The value of M should be as small as possible. **(5)**

Solution By the prime number theorem, the number of primes $\leqslant x$ is nearly $x/\ln x$, that is, the probability that a randomly chosen integer $\leqslant x$ is prime is nearly $1/\ln x$. Under the assumption that x and $2x+1$ both behave as random integers, the probability that one $n+i$ is a Sophie German prime is nearly $1/[\ln(n+M)\ln(2(n+M)+1)]$ which is approximately $1/\ln^2 n$. Therefore, we should take $M = \ln^2 n$ (or a small multiple of $\ln^2 n$).

(b) Describe a sieve to throw away the values of $n+i$ for which either $n+i$ or $2(n+i)+1$ has a prime divisor $\leqslant p_t$. Take t as a constant (like 100). **(10)**

Solution We use an array A indexed by i in the range $0 \leqslant i \leqslant M$. It is not essential to know the exact factorizations of $n+i$. Detecting only that $n+i$ or $2(n+i)+1$ is divisible by any p_j suffices to throw away $n+i$.

In view of this, we initialize each array location A_i to 1. **(1)**

Now, take $q = p_j$ for some $j \in \{1, 2, \ldots, t\}$. The condition $q \mid (n+i)$ implies $i \equiv -n \pmod q$, so we set $A_i = 0$ for all values of i satisfying this congruence. **(4)**

Moreover, for $q \neq 2$, the condition $q \mid 2(n+i)+1$ implies $i \equiv -n-2^{-1} \pmod q$, that is, we set $A_i = 0$ for all values of i satisfying this second congruence. **(4)**

After all primes $p_1, p_2, \ldots, p_t$ are considered, we check the primality of $n+i$ and $2(n+i)+1$ only for those i for which we continue to have $A_i = 1$. **(1)**

(c) Describe the gain in the running time, that you achieve using the sieve. **(10)**

Solution Let $P = p_1 p_2 \cdots p_t$ and $Q = p_2 p_3 \cdots p_t$. The probability that a random $n+i$ is not divisible by any p_j is about $\phi(P)/P$. Likewise, the probability that a random $2(n+i)+1$ is not divisible by any p_j is about $\phi(Q)/Q$. Let us assume that the two events "divisibility of $n+i$ by p_j" and "divisibility of $2(n+i)+1$ by p_j" are independent. Then, we check the primality of $n+i$ and $2(n+i)+1$ for about $(M+1)\frac{\phi(P)\phi(Q)}{PQ}$ values of i. Therefore, the speedup obtained is close to $\frac{PQ}{\phi(P)\phi(Q)}$. For $t = 10$, this speedup is about 20; for $t = 100$, it is about 64; and for $t = 1000$, it is about 128. Note that for a suitably chosen t, we may neglect the sieving time which is $O(t + M\log t)$, that is, $O(t + (\log^2 n)(\log t))$. In contrast, each primality test (like Miller-Rabin) takes time $O(\log^3 n)$.

2 In Floyd's variant of Pollard's rho method for factoring the integer n, we compute the values of x_k and x_{2k} and then $\gcd(x_k - x_{2k}, n)$, for $k = 1, 2, 3, \ldots$. Suppose that we instead compute x_{rk+1} and x_{sk} and subsequently $\gcd(x_{rk+1} - x_{sk}, n)$, for $k = 1, 2, 3, \ldots$, where $r, s \in \mathbb{N}$.

(a) Deduce a condition relating r, s and the length l of the cycle such that this method is guaranteed to detect a cycle of length l. **(10)**

Solution A cycle of length l is detected if and only if $rk + 1 \equiv sk \pmod{l}$ (but $rk + 1 \neq sk$ as integers) for all sufficiently large k. This condition is equivalent to $(r - s)k \equiv -1 \pmod{l}$. This congruence has a solution if and only if $\gcd(r - s, l) = 1$.

(**Remark:** Without the $+1$ in the first walk, any cycle will be detected as long as $r \neq s$. This is because we now require $rk \equiv sk \pmod{l}$, that is, $(r - s)k \equiv 0 \pmod{l}$. This congruence is solvable for k for any value of r and s. The condition $rk \neq sk$ (as integers) demands $r \neq s$.)

(b) Characterize all the pairs (r, s) such that this method is guaranteed to detect cycles of any length. **(5)**

Solution The condition $\gcd(r - s, l) = 1$ for all positive integers l is satisfied if and only if $r - s = \pm 1$.

3 Dixon's method for factoring an integer n can be combined with a sieve in order to reduce its running time to $L[3/2]$. Instead of choosing random values of $x_1, x_2, \ldots, x_s$ in the relations, we first choose a random value of x and, for $-M \leqslant c \leqslant M$, we check the smoothness of the integers $(x + c)^2 \pmod{n}$ over t small primes $p_1, p_2, \ldots, p_t$. As in Dixon's original method, take $t = L[1/2]$.

(a) Determine the value of M for which one expects to get a system of the desired size. **(5)**

Solution For a randomly chosen x, the integer $T(c) = (x + c)^2 \operatorname{rem} n$ is of value $O(n)$ and so has a probability of $L\left[\frac{-1}{2 \times \frac{1}{2}}\right] = L[-1]$ of being $L[1/2]$-smooth. That is, $L[1]$ values of c need to be tried in order to obtain a single relation. Since we require about $2t$ (which is again $L[1/2]$) relations, the value of M should be $L[1] \times L[1/2] = L[3/2]$.

(b) Describe a sieve over the interval $[-M, M]$ for detecting the smooth values of $(x + c)^2 \pmod{n}$. **(10)**

Solution Follow a strategy similar to the QSM. Let $x^2 = kn + J$ with $J \in \{0, 1, 2, \ldots, n-1\}$. We have $(x + c)^2 \equiv x^2 + 2xc + c^2 \equiv kn + J + 2xc + c^2 \equiv T(c) \pmod{n}$, where $T(c) = J + 2xc + c^2$. **(2)**

Use an array A indexed by c in the range $-M \leqslant c \leqslant M$. Initialize $A_c = \log|T(c)|$. **(2)**

For each small prime q and small exponent h, solve the congruence $(x + c)^2 \equiv kn \pmod{q^h}$. For all values of c in the range $-M \leqslant c \leqslant M$, that satisfy the above congruence, subtract $\log q$ from A_c. **(5)**

When all q and h values are considered, check which array locations A_c store values close to 0. Perform trial divisions on the corresponding $T(c)$ values. **(1)**

(c) Deduce how you achieve a running time of $L[3/2]$ using this sieve. **(10)**

Solution Follow the analysis of sieving in QSM. Initializing A takes $L[3/2]$ time. Solving all the congruences $(x + c)^2 \equiv kn \pmod{q^h}$ takes $L[1/2]$ time. Subtraction of all $\log q$ values takes $L[3/2]$ time. Trial division of $L[1/2]$ smooth values by $L[1/2]$ primes takes $L[1]$ time. Finally, the sparse system with $L[1/2]$ variables and $L[1/2]$ equations can be solved in $L[1]$ time. **(2×5)**

4 (a) Let $h \in \mathbb{F}_q^*$ have order m (a divisor of $q - 1$). Prove that for $a \in \mathbb{F}_q^*$, the discrete logarithm $\operatorname{ind}_h a$ exists if and only if $a^m = 1$. **(10)**

Solution Let $a = h^k$ for some $k \in \{0, 1, 2, \ldots, m - 1\}$. Since $\operatorname{ord}(h) = m$, we have $\operatorname{ord}(a) = m / \gcd(m, k)$, that is, $\operatorname{ord}(a) \mid m$, that is, $a^m = 1$. **(5)**

The equation $x^m - 1$ has m (distinct) roots h^k for $k = 0, 1, 2, \ldots, m - 1$. Since $\mathbb{F}_q$ is a field, the polynomial $x^m - 1$ cannot have more than m roots, that is, $a^m = 1$ implies that $a = h^k$ for some k. **(5)**

(b) Suppose that g and g' are two primitive elements of $\mathbb{F}_q^*$. Show that if one can compute discrete logarithms to the base g in $O(f(\log q))$ time, then one can also compute discrete logarithms to the base g' in $O(f(\log q))$ time. (You may assume that $f(\log q)$ is a super-polynomial expression in $\log q$.) **(10)**

Solution Let $g' = g^r$ for some r with $\gcd(r, q-1) = 1$. **(3)**

Take $a = (g')^x = g^{rx}$. Then, $\operatorname{ind}_{g'} a \equiv x \equiv r^{-1} \times (rx) \equiv r^{-1} \operatorname{ind}_g a \pmod{q-1}$. But $r = \operatorname{ind}_g g'$, that is, $\operatorname{ind}_{g'} a \equiv (\operatorname{ind}_g g')^{-1} \times (\operatorname{ind}_g a) \pmod{q-1}$. **(5)**

In other words, two index calculations to the base g give $\operatorname{ind}_{g'} a$. The total effort of two index calculations is $O(f(\log q))$. The additional effort associated with one inverse and one multiplication modulo $q-1$ requires $o(f(\log q))$ time. **(2)**

5 Suppose that in the linear sieve method for computing discrete logarithms in $\mathbb{F}_p$, we obtain an $m \times n$ system of congruences, where $n = t + 2M + 2$ and $m = 2n$. Assume that the $T(c_1, c_2)$ values behave as random integers (within a bound). Calculate the expected number of non-zero entries in the $m \times n$ coefficient matrix. You may make use of the fact that, for a positive real number x, the sum of the reciprocals of the primes $\leqslant x$ is approximately $\ln \ln x + \mathcal{B}_1$, where $\mathcal{B}_1 = 0.2614972128\ldots$ is known as the *Mertens constant*. (Note that the expected number of non-zero entries is significantly smaller than the obvious upper bound $O(m \log p)$.) **(15)**

Solution Number the columns of the coefficient matrix A by $0, 1, 2, \ldots, t + 2M + 1$. Column 0 corresponds to the "prime" -1, Columns 1 through t to the small primes $p_1, p_2, \ldots, p_t$, and Columns $t+1$ through $t + (2M+1)$ to the $H + c$ values for $-M \leqslant c \leqslant M$. Suppose also that the last row corresponds to the *free* relation $\operatorname{ind}_g(p_j) = 1$ for some j. This row has only one non-zero entry. We now count the number of non-zero entries in the first $m-1$ rows.

The expected number of non-zero entries in Column 0 is $(m-1)/2$. **(2)**

For $1 \leqslant j \leqslant t$, the expected number of non-zero entries in Column j is $(m-1)/p_j$, since a randomly chosen integer is divisible by the prime p_j with probability $1/p_j$. **(5)**

Finally, consider the submatrix consisting of the first $m-1$ rows and the last $2M+1$ columns. Each row in this submatrix has exactly two non-zero entries corresponding to the two values c_1, c_2 for a smooth $T(c_1, c_2)$. Of course, we allow the possibility $c_1 = c_2$ during sieving (in which case there is only one non-zero entry in a row), but this situation occurs with a low probability, and we expect to get at most only a small constant number of such rows. In view of this, we neglect the effects of these rows in our final count. **(5)**

To sum up, the expected number of non-zero entries in A is nearly

$$1 + (m-1)/2 + (m-1)\left(\sum_{j=1}^{t} \frac{1}{p_j}\right) + 2(m-1).$$

By the prime number theorem, the t-th prime p_t is approximately equal to $t \ln t$, and so the sum $\sum_{j=1}^{t} \frac{1}{p_j}$ equals $\ln \ln(t \ln t) + \mathcal{B}_1$, approximately. Combining these observations, we conclude that the expected count of non-zero entries in A is nearly **(3)**

$$\begin{aligned} & 1 + (m-1)\Big(\ln \ln(t \ln t) + \mathcal{B}_1 + 5/2\Big) \\ = \; & 1 + (2t + 4M + 3)\Big(\ln \ln(t \ln t) + \mathcal{B}_1 + 5/2\Big). \end{aligned}$$

(This estimate indicates that we expect only $\Theta(\ln \ln t)$ non-zero entries per row, on an average. Since $t = L[1/2]$, this count is $\Theta(\ln \ln p)$—a quantity exponentially tighter than the obvious upper bound $O(\log p)$.)