Computational Number Theory (CS60094)

Spring Semester, 2023

Tutorial - 3

1. For a positive integer n, the sum of the reciprocals of all primes $\leq n$ asymptotically approaches $\ln \ln n$. Using this fact, derive that the sieve of Eratosthenes can be implemented to run in $\mathcal{O}(n \ln \ln n)$ time.

2. Modify the sieve of Eratosthenes so that it runs in $\mathcal{O}(n)$ time.

3. If both p and $(2p+1)$ are prime, we call p a Sophie-Germaine prime. Locate the smallest Sophie-Germaine prime $p \geq n$ for a given positive integer $n >> 1$, by sieving over interval $[n, n+M]$.
   - Find a value of M such that there is at least one Sophie-Germaine prime in the interval $[n, n+M]$ with high probability.
   - Describe the sieving mechanism to throw away the values of $(n+i)$ for which either $(n+i)$ or $2(n+i)+1$ has a prime divisor less than or equal to the t^{th} prime, where t is some constant.

4. Let s and t be bit length with $s > t$.
   - Describe an algorithm to locate a random s-bit prime p such that a random prime of bit length t divides $(p-1)$.
   - Analyze its runtime.

5. Prove the following properties of any Carmichael number n.
   - $(p-1)|(n-1)$, for every prime divisor p of n.
   - n is odd.
   - n is square-free.

6. Suppose that A_y is a yes-biased algorithm for proving the primality of an integer and A_n is a no-biased algorithm for the same purpose. Prove or disprove: by running A_y and A_n alone, we can deterministically conclude about the primality of an integer.

7. Let $n \in \mathbb{N}$ be odd and composite. If n is not a pseudoprime to some base in $\mathbb{Z}_n^*$, prove that n is not a pseudoprime to at least half of the bases in $\mathbb{Z}_n^*$.

8. Let $n \in \mathbb{N}$ be odd and composite. If n is not a Euler pseudoprime to some base in $\mathbb{Z}_n^*$, prove that n is not a Euler pseudoprime to at least half of the bases in $\mathbb{Z}_n^*$.

9. Write the binary search algorithm for finding k^{th} root of an integer.