CS-E4500 Problem Set 9

Jaan Tollander de Balsch - 452056

March 29, 2019

Material used in this report: Gathen and Gerhard (2013), Sections 19.1-3, 19.5.

Problem 1

Factor the integer $N = 2028455971$.

Given s and t such that $s^2 \equiv t^2 \mod N$ and $\gcd(s, N) = \gcd(t, N) = 1$ there exists integer q with

$$s^2 - t^2 = (s - t)(s + t) = qN.$$

Thus

$$\gcd(s + t, N)$$

is a proper divisor of N unless N divides $s - t$ or N divides $s + t$.

We are given $s = 702505371$ and $t = 188270011$ then

$$\gcd(s + t, N) = 46073$$

is prime and a factor of N. Dividing N by this number gives

$$N/46073 = 44027,$$

which is also a prime and a factor of N. Therefore the factors of N are 44027 and 46073.

Problem 2

De Bruijn's lower bound. Show that for all $x \geq 1$ and $k \in \mathbb{Z}_{\geq 1}$ we have $\Psi(x^k, x) \geq \binom{\pi(x)+k}{k} \geq \left(\frac{\pi(x)}{k}\right)^k$.

Let $\Psi(M, B)$ be the number of B-smooth positive integers N at most M. Let $\pi(x)$ denote the number of prime numbers at most x.

The factorization of B-smooth positive integer N at most M is

$$N = p_1^{z_1} p_2^{z_2} \cdots p_r^{z_r} \leq M$$

where $p_1, p_2, ..., p_r \leq B$ are its distinct primes factors, $z \in \mathbb{Z}_{\geq 0}$ the coefficients and $r = \pi(B)$ the number of distinct prime factors.

Using the definition

$$\begin{aligned} N &= p_1^{z_1} p_2^{z_2} \cdots p_r^{z_r} \\ &\leq B^{z_1} B^{z_2} \cdots B^{z_r} \\ &= B^{z_1 + z_2 + \ldots + z_r} \\ &= M. \end{aligned}$$

Substituting $M = x^k$ and $B = x$ and $r = \pi(x)$ gives

$$\begin{aligned} x^{z_1 + z_2 + \ldots + z_r} &= x^k \\ z_1 + z_2 + ... + z_r &= k. \end{aligned}$$

The amount of solutions to this inequality is

$$\binom{r+k}{k} = \binom{\pi(x)+k}{k}.$$

Giving us the lower bound of

$$\Psi(x^k, x) \geq \binom{\pi(x)+k}{k}.$$

Using the definition for the binomial

$$\binom{m}{k} = \frac{m!}{k!\,(m-k)!}$$

and for the factorial

$$m! = 1 \cdot 2 \cdots m \leq m^m$$

the lower bound can be furher refined into

$$\begin{aligned} \binom{r+k}{k} &= \frac{(r+k)!}{k!\,((r+k)-k)!} \\ &= \frac{(r+k)!}{k! \cdot r!} \\ &= \frac{(r+1)(r+2)\cdots(r+k)}{1 \cdot 2 \cdots k} \\ &\geq \frac{r^k}{k^k} \\ &= \left(\frac{r}{k}\right)^k. \end{aligned}$$

Therefore

$$\Psi(x^k, x) \geq \binom{\pi(x)+k}{k} \geq \left(\frac{\pi(x)}{k}\right)^k.$$

Problem 3

Given an integer $N \in \mathbb{Z}_{\geq 2}$ as input, design an algorithm that either

1) outputs a prime p and a positive integer a such that $N = p^a$ or
2) asserts that N is not a prime power.

Your algorithm should run in time $O((\log N)^c)$ for a constant $c > 0$. Carefully justify the running time of your algorithm. You may assume that you have available a subroutine that tests whether a given $m \in \mathbb{Z}_{\geq 2}$ is prime in time $O((\log m)^d)$ for a constant $d > 0$.

The full algorithm Find-Prime-Power(N) has two main subroutines Is-Prime(n) and Find-kth-Power-Less-Eq-Than(B, k, N) which are explained and analyzed below.

The subroutine Is-Prime(n) tests whether integer n is prime in $O((\log m)^d)$ time.

Input: Monotonically increasing sequence $B \in \mathbb{Z}_{\geq 2}^n$ of length n, an integer $k \in \mathbb{Z}_{\geq 1}$ and an integer $N \in \mathbb{Z}_{\geq 2}$.

Output: Largest element $b \in B$ such that $b^k \leq N$. If no such elements exists output NIL.

Find-kth-Power-Less-Eq-Than(B, k, N)

1) **return** largest element $b \in B$ such that $b^k \leq N$ using (modified) *binary search* or NIL if no such b exists.

Analysis: The worst case performance of the binary search is $O(\log n)$ iterations.

Input: An integer $N \in \mathbb{Z}_{\geq 2}$.

Output: A tuple (b, k) where b is a prime and k is a positive integer such that $N = b^k$. If no such number exists outputs NIL.

Find-Prime-Power(N)

1) $k \leftarrow 0$
2) $b \leftarrow N$
3) **while** $True$
4) ..... $B \leftarrow \langle 2, 3, ..., b \rangle$

5) ..... $k \leftarrow k + 1$
6) ..... $b \leftarrow$ Find-kth-Power-Less-Eq-Than(B, k, N)
7) ..... **if** $b = NIL$
8) .... **return** NIL
9) ..... **if** $b^k = N$ and Is-Prime(b)
10) .... **return** (b, k)

Correctness: Proof that the set of possible prime bases B for degree k can be limited to b. Let $b^k \leq N$. Then for all $d > b$ we have

$$N < d^k < d^{k+1}.$$

Therefore $d > b$ cannot be a prime base for degree $k + 1$ and the set B can be limited to b from previous iteration.

Analysis: The progression of the value of b towards values 2 determines the number of iterations within the while loop. It follow the sequence

$$\lfloor N^1 \rfloor, \lfloor N^{1/2} \rfloor, \lfloor N^{1/3} \rfloor, ..., 2.$$

The number of elements in the sequence is given by solving the smallest positive integer i such $\lfloor N^{1/i} \rfloor = 2$.

$$\begin{aligned} \lfloor N^{1/i} \rfloor &= 2 \\ 2 &\leq N^{1/i} < (2+1) \\ 2^i &\leq N < 3^i \\ \log_3 N &< i \leq \log_2 N \end{aligned}$$

Therefore the while loop has $O(\log N)$ iterations.

Total number of operations consists of the number of iterations in the while loop multiplied by the sum of the number of operations in the primality test and finding k-th power less or equal than using binary search

$$O(\log N) \cdot (O((\log m)^d) + O(\log n)) \in O(\log N)^c$$

where $c > 0$ is a constant, $n \leq M \leq N$, $d < c$ and $m \leq N$.

Problem 4

References

Gathen, Joachim von zur, and Jurgen Gerhard. 2013. *Modern Computer Algebra*. 3rd ed. New York, NY, USA: Cambridge University Press.