CSE 373
HW 1

David Lai

Question 1

Figure 1 below shows a non-recursive loop-based algorithm.

(a) What computational problem does the algorithm solve? Justify your answer.

(b) Give a tight bound on the worst-case running time of the algorithm. Show your analysis.

(c) Suppose that we replace the statement in line 5 with $j \leftarrow j + i^2$. Now give a tight bound on the worst case running time of this modified version of the algorithm. Show your analysis.

Solution:

(a) This algorithm generates primes in the range $[1 : n]$. We can see this as lines 1-8 are used to count the number of divisors for each index i. It does this by looping over the divisor j and incrementing the values at any indices that are a multiple of j. Then, in lines 9-12, the algorithm prints out all indices which have a value of zero, indicating that the index has no divisors other than one and itself.

(b) We can define the worst-case running time of this algorithm as:

$$T(n) = c_1 n + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i} + c_4(n-1)$$

We can begin by solving for the upper bound.

$$T(n) \leq c_1 n + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i} + c_4(n-1)$$

Because $n - 1 < n$,

$$T(n) \leq c_1 n + c_2(n) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i} + c_4(n)$$

Using $n(c_1 + c_2 + c_4) = \Theta(n)$,

$$T(n) \leq \Theta(n) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i}$$

Splitting the sum,

$$T(n) \leq \Theta(n) + c_3 n \sum_{i=2}^{n} \frac{1}{i} - c_3 \sum_{i=2}^{n} 2$$
$$T(n) \leq \Theta(n) + c_3 n \sum_{i=2}^{n} \frac{1}{i} - 2c_3(n-1)$$

Expanding the sum,

$$T(n) \leq \Theta(n) + c_3 n(\frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}) - 2c_3(n-1)$$
$$T(n) \leq \Theta(n) + c_3 n(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}) - c_3 n - 2c_3 n$$

Since the harmonic series $H_n = \Theta(\log n)$,

$$T(n) \leq \Theta(n) + c_3 n[\Theta(\log n) - 3]$$
$$T(n) \leq \Theta(n) + \Theta(n \log n) - 3c_3 n$$
$$T(n) \leq \Theta(n \log n)$$
$$\implies T(n) = O(n \log n)$$

Now, solving for the lower bound:

$$T(n) \geq c_1 n + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i} + c_4(n-1)$$

Because $n - 1 < n$,

$$T(n) \geq c_1(n-1) + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i} + c_4(n-1)$$

Using $(n-1)(c_1 + c_2 + c_4) = \Theta(n)$,

$$T(n) \geq \Theta(n) + c_3 \sum_{i=2}^{n} \frac{n-2i}{i}$$

Splitting the sum,

$$T(n) \geq \Theta(n) + c_3 n \sum_{i=2}^{n} \frac{1}{i} - c_3 \sum_{i=2}^{n} 2$$

$$T(n) \geq \Theta(n) + c_3 n \sum_{i=2}^{n} \frac{1}{i} - 2c_3(n-1)$$

Expanding the sum,

$$T(n) \geq \Theta(n) + c_3 n(\frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}) - 2c_3(n-1)$$

$$T(n) \leq \Theta(n) + c_3 n(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}) - c_3 n - 2c_3(n-1)$$

Since the harmonic series $H_n = \Theta(\log n)$,

$$T(n) \geq \Theta(n) + c_3 n[\Theta(\log n)] - 3c_3(n-1)$$

$$T(n) \geq \Theta(n) + \Theta(n \log n)$$

$$T(n) \geq \Theta(n \log n)$$

$$\implies T(n) = \Omega(n \log n)$$

Combining our upper and lower bounds, we see that $T(n) = \Theta(n \log n)$, where $T(n)$ is the worst case running time of the algorithm.

(c) Our new running time becomes:

$$T(n) = c_1 n + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n-2i^2}{i^2} + c_4(n-1)$$

Solving for our upper bound:

$$T(n) \leq c_1 n + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n-2i^2}{i^2} + c_4(n-1)$$

Because $n - 1 < n$,

$$T(n) \leq c_1 n + c_2(n) + c_3 \sum_{i=2}^{n} \frac{n-2i^2}{i^2} + c_4(n)$$

Using $n(c_1 + c_2 + c_4) = \Theta(n)$,

$$T(n) \leq \Theta(n) + c_3 \sum_{i=2}^{n} \frac{n-2i^2}{i^2}$$

Splitting the sum,

$$T(n) \leq \Theta(n) + c_3 n \sum_{i=2}^{n} \frac{1}{i^2} - c_3 \sum_{i=2}^{n} 2$$

Expanding the sum,

$$\begin{aligned}
T(n) &\leq \Theta(n) + c_3 n(\frac{1}{4} + \frac{1}{9} + \ldots + \frac{1}{n^2}) - 2c_3(n-1) \\
T(n) &\leq \Theta(n) + c_3 n(1 + \frac{1}{4} + \frac{1}{9} + \ldots + \frac{1}{n^2}) - c_3 n - 2c_3(n)
\end{aligned}$$

Since the series $\sum_{i=1}^{n} \frac{1}{i^2} = \frac{\pi^2}{6}$,

$$\begin{aligned}
&T(n) \leq \Theta(n) + c_3 n[\Theta(1)] - 3c_3(n) \\
&T(n) \leq \Theta(n) + \Theta(n) - \Theta(n) \\
&T(n) \leq \Theta(n) \\
&\implies T(n) = O(n)
\end{aligned}$$

Now, solving for the lower bound:

$$T(n) \geq c_1 n + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n - 2i^2}{i^2} + c_4(n-1)$$

Because $n - 1 < n$,

$$T(n) \geq c_1(n-1) + c_2(n-1) + c_3 \sum_{i=2}^{n} \frac{n - 2i^2}{i^2} + c_4(n-1)$$

Using $(n-1)(c_1 + c_2 + c_4) = \Theta(n)$,

$$T(n) \geq \Theta(n) + c_3 \sum_{i=2}^{n} \frac{n - 2i^2}{i^2}$$

Splitting the sum,

$$T(n) \geq \Theta(n) + c_3 n \sum_{i=2}^{n} \frac{1}{i^2} - c_3 \sum_{i=2}^{n} 2$$

Expanding the sum,

$$\begin{aligned}
T(n) &\geq \Theta(n) + c_3 n(\frac{1}{4} + \frac{1}{9} + \ldots + \frac{1}{n^2}) - 2c_3(n-1) \\
T(n) &\geq \Theta(n) + c_3 n(1 + \frac{1}{4} + \frac{1}{9} + \ldots + \frac{1}{n^2}) - c_3 n - 2c_3(n-1)
\end{aligned}$$

Since the series $\sum_{i=1}^{n} \frac{1}{i^2} = \frac{\pi^2}{6}$,

$$\begin{aligned}
&T(n) \geq \Theta(n) + c_3 n[\Theta(1)] - 3c_3(n-1) \\
&T(n) \geq \Theta(n) + \Theta(n) - \Theta(n) \\
&T(n) \geq \Theta(n) \\
&\implies T(n) = \Omega(n)
\end{aligned}$$

Combining our upper and lower bounds, we see that $T(n) = \Theta(n)$, where $T(n)$ is the worst case running time of the algorithm.

Question 2

Performance of which aspect/part of the original insertion sort algorithm does Improved-Insertion-Sort try to improve? How and why does this improved version work? Give a clear and detailed explanation.

Solution: