Integer Logarithm by Bit-Twidding

Borderite · 7 Oct 2026

https://functor.network/user/3329/entry/2003

Integer Logarithm by Bit-Twidding

Borderite

October 9, 2026

This note considers computation of the integer logarithm, i.e., the logarithm truncated down to the nearest integer. If the base of the logarithm is two, the integer logarithm is easily obtained by counting the number of bits right to the most significant set bit of the number. We thus focus on the case with the base greater than two.

Let b and W be natural numbers such that $b \geq 3$. We treat them number as constants in our analysis unless otherwise stated. Also, write

$$\mathcal{N}_W \equiv \{n \in \mathbb{N} : \, 0 \leq n \leq 2^W - 1\}.$$

In this note, we investigate computation of the integer logarithm with base b:

$$q(n) \equiv \lfloor \log_b n \rfloor$$

by using

$$\hat{q}_k(n) \equiv \left\lfloor \frac{m_k \cdot \lfloor \log_2 n \rfloor}{2^k} \right\rfloor,$$

where k is a suitably chosen natural number, $m_k \equiv \lfloor 2^k \log_b 2 \rfloor$. This approach is motivated by the simple identity:

$$\log_b n = \log_b 2 \cdot \log_2 n, \quad n \in \mathcal{N}_W \backslash \{0\}.$$

We attempt to approximate $\log_b 2$ by $m_k / 2^k$ and $\log_2 n$ by $\lfloor \log_2 n \rfloor$. Chapter 11 of (Henry S. Warren 2013) studies this very approach for the case with $b = 10$. This note provides a natural extension to arbitrary bases along with some detailed mathematical analysis.

The sequence $\{m_k / 2^k\}_{k \in \mathbb{N}}$ is bounded and nondecreasing, because

$$\frac{m_k}{2^k} = \frac{\lfloor 2^k \log_b 2 \rfloor}{2^k} \leq \frac{2^k \log_b 2}{2^k} = \log_b 2,$$

and

$$\begin{aligned}\frac{m_k}{2^k} &= \frac{\lfloor 2^k \log_b 2 \rfloor}{2^k} = \frac{\lfloor 2 \cdot 2^{k-1} \log_b 2 \rfloor}{2^k} \geq \frac{\lfloor 2 \cdot \lfloor 2^{k-1} \log_b 2 \rfloor \rfloor}{2^k} \\ &= \frac{2 \cdot \lfloor 2^{k-1} \log_b 2 \rfloor}{2^k} = \frac{m_{k-1}}{2^{k-1}}.\end{aligned}$$

It actually converges to $\log_b 2$, since

$$0 \leq \log_b 2 - \frac{m_k}{2^k} \leq \frac{1}{2^k}(2^k \log_b 2 - \lfloor 2^k \log_b 2 \rfloor) \leq \frac{1}{2^k} \to 0 \text{ as } k \to \infty. \tag{1}$$

Thus, we can approximate $\log_b 2$ by $m_k/2^k$ arbitrarily well by using a k sufficiently large. Nevertheless, $q(n)$ and $\hat{q}_k(n)$ can still disagree even with a very large k. For example, suppose that n is not a power of two and that $n = b$. Then for any $k \in \mathbb{N}$,

$$\begin{aligned}\hat{q}_k(b) &= \left\lfloor \frac{m_k}{2^k} \lfloor \log_2 b \rfloor \right\rfloor \leq \frac{m_k}{2^k} \lfloor \log_2 b \rfloor \leq \log_b 2 \cdot \lfloor \log_2 b \rfloor \\ &< \log_b 2 \cdot \log_2 b = \log_b b = 1 = q(b),\end{aligned}$$

where the strong inequality holds because $\log_2 b$ is not a whole number. For this reason, we only aim to formulate $\hat{q}_k(n)$ that delivers either $q(n)$ or $q(n) - 1$ for each $n \in \mathcal{N}_W \backslash \{0\}$.

Fix $k \in \mathbb{N}$. Because

$$\log_b n = \log_b 2 \cdot \log_2 n \geq \frac{m_k}{2^k} \cdot \lfloor \log_2 n \rfloor,$$

it holds that

$$q(n) = \lfloor \log_b n \rfloor \geq \left\lfloor \frac{m_k}{2^k} \cdot \lfloor \log_2 n \rfloor \right\rfloor = \hat{q}_k(n). \tag{2}$$

Define

$$\epsilon_k \equiv \log_b 2 - \frac{m_k}{2^k}.$$

Then it also holds that

$$\begin{aligned}q(n) &\leq \lfloor \log_b n \rfloor = \lfloor \log_b 2 \cdot \log_2 n \rfloor = \left\lfloor \left(\frac{m_k}{2^k} + \epsilon_k \right) \log_2 n \right\rfloor \\ &= \left\lfloor \frac{m_k}{2^k} \cdot \log_2 n + \epsilon_k \log_2 n \right\rfloor = \left\lfloor \frac{m_k}{2^k} \cdot \log_2 n + \epsilon_k \log_2 n \right\rfloor \\ &\leq \left\lfloor \frac{m_k}{2^k} (\lfloor \log_2 n \rfloor + 1) + \epsilon_k \log_2 n \right\rfloor = \left\lfloor \frac{m_k}{2^k} \lfloor \log_2 n \rfloor + \frac{m_k}{2^k} + \epsilon_k \log_2 n \right\rfloor \\ &\leq \left\lfloor \frac{m_k}{2^k} \lfloor \log_2 n \rfloor \right\rfloor + \left\lfloor \frac{m_k}{2^k} + \epsilon_k \log_2 n \right\rfloor + 1 \\ &= \left\lfloor \frac{m_k}{2^k} \lfloor \log_2 n \rfloor \right\rfloor + \lfloor \log_b 2 - \epsilon_k + \epsilon_k \log_2 n \rfloor + 1 \\ &= \left\lfloor \frac{m_k}{2^k} \lfloor \log_2 n \rfloor \right\rfloor + \lfloor \log_b 2 + \epsilon_k (\log_2 n - 1) \rfloor + 1 = \hat{q}_k(n) + E_k(n),\end{aligned}$$

where $E_k : \mathbb{N} \to \mathbb{R}$ is defined by

$$E_k(n) \equiv \lfloor \log_b 2 + \epsilon_k(\log_2 n - 1) \rfloor + 1$$

It follows that

$$\hat{q}_k(n) \geq q(n) - E_k(n) \tag{3}$$

Combining (2) and (3), we obtain that

$$q(n) - E_k(n) \leq \hat{q}_k(n) \leq q(n).$$

Note that $\hat{q}_k(n)$ and $q(n)$ are integers. If $E_k(n) < 2$, it is guaranteed that $\hat{q}_k(n) = q(n)$ or that $\hat{q}_k(n) = q(n) - 1$. Because $E_k(n)$ is nondecreasing in n, we have that for each $n \in \mathcal{N}_W \backslash \{0\}$,

$$E_k(n) \leq \lfloor \log_b 2 + \epsilon_k(\log_2 2^W - 1) \rfloor + 1 = \lfloor \log_b 2 + \epsilon_k(W - 1) \rfloor + 1$$

It follows by (1) that

$$E_k(n) \leq \left\lfloor \log_b 2 + \frac{W-1}{2^k} \right\rfloor + 1.$$

The sequence $\{\log_b 2 + (W-1)/2^k\}_{k \in \mathbb{N}}$ converges to $\log_b 2$, which is less than one, provided that $b > 2$. It follows that $E_k(n)$ converges to one uniformly in $n \in \mathcal{N}_W \backslash \{0\}$ as $k \to \infty$. Thus, there always exists $k \in \mathbb{N}$ such that $E_k(n) < 2$ for every $n \in \mathcal{N}_W \backslash \{0\}$.

Once we choose a suitable k, we need to determine which $\hat{q}_k(n)$ is equal to, $q(n)$ or $q(n) - 1$. To this end, we can use the fact that $\hat{q}_k(n) = q(n)$ if and only if $b^{\hat{q}_k(n)+1} > n$. To avoid the relatively expensive computation of $b^{\hat{q}_k(n)+1}$, we can use a table lookup scheme. That is, we make in our code an array that maps x to b^{x+1} for each x in $\{0, 1, \ldots, \lfloor \log_b(2^W - 1) \rfloor - 1\}$ in an array in the code. This allows us to quickly obtain the value of $b^{\hat{q}_k(n)+1}$. When $\hat{q}_k(n) = \lfloor \log_b(2^W - 1) \rfloor$, we are sure that $\hat{q}_k(n) = q(n)$, because $b^{\hat{q}_k(n)+1} \notin \mathcal{N}_W$.

Example 1. *Suppose that $b = 10$ and $W = 32$. Then the value of $\bar{E}_k \equiv E_k(2^W - 1)$ becomes 1 when $k = 5$.*

k	1	2	3	4	5	6	7	8	9
m_k	0	1	2	4	9	19	38	77	154
$E_k(2^W - 1)$	10	2	2	2	1	1	1	1	1

Thus,

$$\hat{q}_5(n) \equiv \left\lfloor \frac{9 \cdot \lfloor \log_2 n \rfloor}{2^5} \right\rfloor$$

is equal to $q(n)$ or $q(n) - 1$ for every $n \in \mathcal{N}_W$. The value of $\lfloor \log_{10}(2^{32} - 1) \rfloor$ is nine. When $\hat{q}_k(n) = 9$, we know that $q(n) = 9$.

Here is a sample implementation for the case with $b = 10$ and $W = 32$ written in C.

```
unsigned int uint32__log10(unsigned int x) {
  static const unsigned int threshold[] = {
```

```
    10, 100, 1000, 10000, 100000, 1000000,
    10000000, 100000000, 1000000000
  };
  unsigned int floor_log2 = 31 - __builtin_clz(x);
  unsigned int y = (floor_log2 * 9) >> 5;
  return y + (x >= threshold[y]);
}
```

Example 2. *Suppose that* $b = 5$ *and* $W = 32$*. Then the value of* $\bar{E}_k \equiv E_k(2^W - 1)$ *becomes 1 when* $k = 6$*.*

k	0	1	2	3	4	5	6	7	8
m_k	0	0	1	3	6	13	27	55	110
$E_k(2^W - 1)$	14	14	7	3	3	2	1	1	1

Thus,

$$\hat{q}_6(n) \equiv \left\lfloor \frac{27 \cdot \lfloor \log_2 n \rfloor}{2^6} \right\rfloor$$

is equal to $q(n)$ *or* $q(n) - 1$ *for every* $n \in \mathcal{N}_W$*. The value of* $\lfloor \log_5(2^{32} - 1) \rfloor$ *is 13. When* $\hat{q}_k(n) = 13$*, we know that* $q(n) = 13$*.*

When we need to calculate the logarithm with various bases, we could not employ the table lookup scheme. We can, however, still use an efficient algorithm for computing b^x such as the exponentiation by squaring.

References

Henry S. Warren, Jr. 2013. *Hacker's Delight*. 2nd edition. Addison Wesley–Pearson Education.