Integer Representations and Arithmetic

Yan Huang
Decimal Numbers

• Examples:
  5
  123
  8920

• Base-10
The Concept of Bases

• Base-16 (Hexadecimal)
  - Ounces and Pounds
  - How many ounces is 32 pounds?

• Base-12
  - Feet and Inches
  - How many inches are there in 6 feet?

• Base-8
  - Gallons and Pints

• Base-4
  - Dollars and Quarters, Gallons and Quarts

• Base-2 (Binary)
Base-10 Representation

• Ten states: 0, 1, ..., 9.
• Need more than one digit to denote numbers greater than or equal to 10.
  - e.g., 10, 11
• 139
• 8901
Binary Numbers

- \( (0)_2 = (0)_{10} \)
- \( (1)_2 = (1)_{10} \)
- \( (10)_2 = (2)_{10} \)
- \( (11)_2 = (3)_{10} \)
- \( (1001)_2 = (9)_{10} \)
- \( (1100)_2 = (12)_{10} \)

\( \text{Octal numbers:} \)

- \( (11)_8 = (7)_{10} \)
- \( (1111)_8 = (15)_{10} \)
Octal Numbers

- $(0)_8 = (0)_10$
- $(1)_8 = (1)_10$
- $(10)_8 = (8)_10$
- $(11)_8 = (9)_10$
- $(1001)_8 = (17)_10$
- $(2671)_8 = (1465)_10$

\[2 \times 512 + 6 \times 64 + 7 \times 8 + 1\]
Hexadecimal

- 16 tokens: 0, 1, ..., 9, A, B, C, D, E, F
- \((A)_{16}\) \(\rightarrow \) \((B)_{16}\) \(\rightarrow \) \((C)_{16}\)
- \((10)_{16} = (16)_{10}\)
- \((11)_{16} = (17)_{10}\)
- \((20)_{16} = (32)_{10}\)
- \((100)_{16} = (1 \times 16^2)_{10} = (256)_{10}\)
- \((3A1)_{16} = (3 \times 16^2 + 10 \times 16 + 1)_{10} = (953)_{10}\)
Base-$b$ Numbers to Decimal Numbers

\[(d_{n-1}d_{n-2} \cdots d_1d_0)_b = \left( d_0 + d_1 \times b + d_2 \times b^2 + \cdots + d_{n-1} \times b^{n-1} \right)_b \]

\[= \sum_{i=0}^{n-1} d_i \times b^i\]
Base-$b$ Numbers to Decimals ($b<10$)

- In Haskell

\[ \text{toDecimal :: Int} \rightarrow [\text{Int}] \rightarrow \text{Int} \]
Base-\textit{b} Numbers to Decimals \textit{(b<10)}
Decimals to Base-$b$ Numbers

\[ d_0 + d_1 \times b + d_2 \times b^2 + \cdots + d_{n-1} \times b^{n-1} = (d_{n-1}d_{n-2} \cdots d_1d_0)_b \]

\[ \left( \frac{123 - d_0}{b} \right) \mod b \]

\[ d_0 = 123 \mod 8 = 3 \]

\[ 120 \mod 8 = 65 \]

\[ 65 \mod 8 = 7 \]

\[ 15 - 7 = 8 \mod 8 = 1 \]

\[ \boxed{x = 8^2 + 7 \times 8 + 3} \]
$1 \times 6^2 + 3 \times 6 + 4 = 36 + 18 + 4 = 58$

$(58)_b = (?,?)_b$  \quad $(134)_b$

$d_0 = 58 \mod 6 = 4$

$d_1 = \left[\frac{(58 - 4)}{16}\right] \mod 6 = 9 \mod 6 = 3$

$d_2 = \left[\frac{(9 - 3)}{16}\right] \mod 6 = 1$

$d_3 = 0$
Decimals to Base-$b$ Numbers (b<10)

toBaseb :: Int -> Int -> [Int]
Decimals to Base-\(b\) Numbers (b<10)
General Cross Base Translations

\((72)_9 = (?)_4\)

\((72)_9 = (7 \times 9 + 2)_{10} = (65)_{10}\)

\[= \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}_4\]
Arbitrary Precision Arithmetic (Addition)
• 9320 + 2938

\[
\begin{array}{c}
9320 \\
+ 2938 \\
\hline
12258
\end{array}
\]
Adding Binary Numbers

• \((100101)_2 + (1101)_2\)
add :: [Int] -> [Int] -> [Int]
a circuit based on Karatsuba-Ofman multiplication [29] of size approximately 8 for adders, the size of the multiplication circuit is improved to 4 input AND gates) and (y constructed according to the "school method" for multiplication, i.e., adding up as shown in Fig. 4 to compute the difference bit.

Fig. 3.

Fig. 4.

Fig. 1.

Fig. 2.

Improved 1-bit Adder (+)

Improved 1-bit Subtractor (−)

Subtraction Circuit (−)

Addition Circuit (+)

Multiplier Circuit (·)

Int

shifted corresponding to the position:

\[ \text{add} :: [\text{Int}] \to [\text{Int}] \to [\text{Int}] \]