이산수학

수업자료 1장

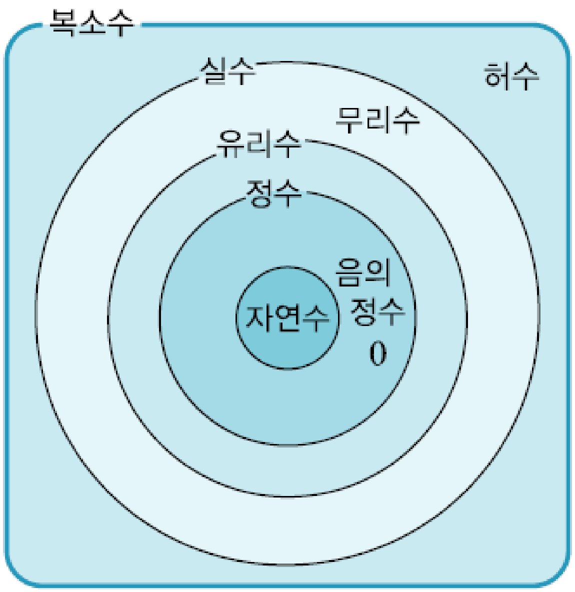
소프트웨어학부
김형균 교수

1장 수의 표현

- 학습목표
  - 수의 체계를 이해한다.
  - 다양한 수의 연산을 이해한다.
  - 다양한 수의 표현을 이해한다.
  - 컴퓨터에서의 수의 표현을 이해한다.
  - 컴퓨터에서의 수의 연산을 이해한다.
- 내용
  - 수 체계
  - 수의 연산
  - 수의 표현
  - 컴퓨터에서의 수의 표현과 연산

1. 수 체계

- 수 체계 : 수의 다양한 형태에 따른 분류
- 수를 표현할 때 파악해야할 것
  - 자릿수 : 소수점을 기준으로 한 위치
  - 기수 : 10진수, 2진수 등과 같이 사용하고 있는 수의 표현 방식



[그림 2-1] 수의 종류

1. 수 체계-자연수

정의 2-1 자연수(Natural Number: N)

0보다 큰 양의 정수

$n, b \in N$이고, $b > 1$, $0 \leq a_i < b$일 때

$$n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b^1 + a_0 b^0 \quad (b: \text{기수},\ k: \text{자릿수})$$

예제 2-1

자연수 589_{10}를 기수와 자릿수를 이용하여 표현하라.

1. 수 체계- 정수

정의 2-2 정수(Integer: Z)

양의 정수, 0, 음의 정수로 구성된 수 체계

예제 2-2

절댓값 $|28|$을 갖는 정수를 모두 표현하라.

1. 수 체계- 유리수

정의 2-3 유리수(Rational Number: Q)

$a, b \in Z$(정수), $a \neq 0$인 경우 $\frac{b}{a}$인 수 체계

- 유리수는 같은 값을 다양하게 표현 가능
- 분자와 분모를 최대공약수로 약분하면 하한항

• 하한항[Lowest]: 분모와 분자 사이에 1 이외의 공약수가 존재하지 않는 유리수

- $\frac{6}{27}$ 은 $\frac{2}{9}$ 가 $\frac{12}{24}$ 는 $\frac{1}{2}$ 이 하한항으로 같은 값

1. 수 체계- 유리수

예제 2-3

다음 유리수가 하한항인지 확인하여 하한항이 아니면 하한항으로 만들어라.

(1) $\dfrac{25}{70}$ (2) $\dfrac{7}{10}$ (3) $\dfrac{9}{14}$ (4) $\dfrac{32}{50}$

1. 수 체계- 무리수

정의 2-4 무리수(Irrational Number: I)

$a, b \in Z$(정수), $a \neq 0$인 경우 $\frac{b}{a}$로 표현할 수 없는 수 체계

- 소수부의 숫자가 유한하거나 무한히 나열되어도 일정한 규칙이 반복되면 유리수
- 소수부의 숫자가 일정한 규칙 없이 무작위로 나열되면 무리수

예제 2-4

다음을 유리수와 무리수로 구분하라.

(1) $\frac{5}{7}$ (2) $\sqrt{2}$ (3) $\sqrt{625}$ (4) $\sqrt{32}$

1. 수 체계- 실수

정의 2-5 실수(Real Number: R)

자연수, 정수, 유리수, 무리수를 모두 포함하는 수 체계

$r \in R$, $b \in N$이고, $b > 1$, $0 \leq a_i < b$일 때

$$r = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b^1 + a_0 b^0 + a_{-1} b^{-1} + a_{-2} b^{-2} + \cdots \quad (k: \text{자릿수})$$

- 실수는 소수점을 기준으로 구분
  - 정수부 : 소수점 왼쪽에 나열된 숫자
  - 소수부 : 오른쪽에 나열된 숫자

예제 2-5

실수 345.734_{10}를 기수와 자릿수를 이용해 표현하라.

1. 수 체계- 복소수

정의 2-6 복소수(Complex Number: C)

$i^2 = -1$을 만드는 허수(i)를 포함하는 수 체계

$c = a + bi$(a: c의 실수부, b: c의 허수부)로 표현되며, 연산은 다음과 같다.

(1) $(a+bi)+(c+di) = (a+c)+(b+d)i$

(2) $(a+bi)(c+di) = ac+adi+bci+bdi^2 = (ac-bd)+(ad+bc)i$

- 복소수를 연산할 때는 실수부는 실수부 끼리, 허수부는 허수부끼리 연산

예제 2-6

복소수 $3+6i$와 $12i$에 대해 합과 곱을 구하라.

2. 수의 연산

- 수의 체계와 연산자의 종류에 따라 연산 결과가 달라짐
  - 수 체계 S 에 속하는 어떤 수 a, b 를 연산자 O 로 연산한 결과 c가 S에 속하면
"S는 연산 O에 대해 닫혀 있다" 라 함

[표 2-1] 수 체계별 사칙연산에 대한 닫힘 성질

	덧셈	뺄셈	곱셈	나눗셈
자연수(N)	○	×	○	×
정수(Z)	○	○	○	×
유리수(Q)	○	○	○	○
무리수(I)	×	×	×	×
실수(R)	○	○	○	○
복소수(C)	○	○	○	○

2. 수의 연산

정리 2-1 **수의 합(덧셈)과 곱(곱셈) 연산의 특징**

(1) 교환법칙: $x + y = y + x$, $xy = yx$

(2) 결합법칙: $(x + y) + z = x + (y + z)$, $(xy)z = x(yz)$

(3) 분배법칙: $x(y + z) = xy + xz$

(4) 합에 대한 항등원: 0

(5) 곱에 대한 항등원: 1

(6) 합에 대한 역: $-a$

(7) 곱에 대한 역: $\frac{1}{a}$

2. 수의 연산-합 연산(Σ)

정의 2-7 합의 표시(Summation: $\sum_{i=1}^{n} a_i$)

일정한 규칙이 있는 수열의 합(i : 합의 색인)

$$\sum_{i=1}^{n} a_i = a_1 + a_2 + \cdots + a_{n-1} + a_n$$

규칙

c는 상수, x_i와 y_i는 실수, k와 n은 양의 정수일 때 규칙은 다음과 같다.

(1) $\sum_{i=1}^{n} c = nc$

증명 $$\sum_{i=1}^{n} c = \underbrace{c + c + c + \cdots + c + c + c}_{n\text{개}} = nc$$

2. 수의 연산-합 연산(Σ)

(2) $\sum_{i=1}^{n} cx_i = c\sum_{i=1}^{n} x_i$

증명 $\sum_{i=1}^{n} cx_i = cx_1 + cx_2 + cx_3 + \cdots + cx_{n-1} + cx_n$

$= c(x_1 + x_2 + x_3 + \cdots + x_{n-1} + x_n) = c\sum_{i=1}^{n} x_i$

2. 수의 연산-합 연산(Σ)

(3) $\sum_{i=1}^{n}(x_i+y_i)=\sum_{i=1}^{n}x_i+\sum_{i=1}^{n}y_i$

증명

$$\begin{aligned}\sum_{i=1}^{n}(x_i+y_i)&=(x_1+y_1)+(x_2+y_2)+\cdots+(x_{n-1}+y_{n-1})+(x_n+y_n)\\&=(x_1+x_2+\cdots+x_{n-1}+x_n)+(y_1+y_2+\cdots+y_{n-1}+y_n)\\&=\sum_{i=1}^{n}x_i+\sum_{i=1}^{n}y_i\end{aligned}$$

2. 수의 연산-합 연산(Σ)

(4) $\sum_{i=1}^{n} x_i = \sum_{i=1}^{k} x_i + \sum_{i=k+1}^{n} x_i$ (단, $1 \le k < n$)

증명 $\sum_{i=1}^{n} x_i = x_1 + x_2 + \cdots + x_k + x_{k+1} + \cdots + x_{n-1} + x_n$

$= (x_1 + x_2 + \cdots + x_k) + (x_{k+1} + \cdots + x_{n-1} + x_n)$

$= \sum_{i=1}^{k} x_i + \sum_{i=k+1}^{n} x_i$

- 예) 1과 20사이에 있는 홀수의 합

2. 수의 연산-합 연산(Σ)

예제 2-7

다음을 연산하라.

(1) $\sum_{i=0}^{4} 2^i$ (2) $\sum_{j=1}^{10} 5$ (3) $\sum_{i=2}^{5} 4i$

2. 수의 연산-합 연산(Σ)

예제 2-7

다음을 연산하라.

(4) $\sum_{k=0}^{3} k^2$

(5) $\sum_{j=1}^{4} j^2 + 9$

(6) $\sum_{j=1}^{4} (j^2 + 9)$

2. 수의 연산-곱 연산(Π와 !)

정의 2-8 곱의 표시(Product: $\prod_{i=1}^{n} a_i$)

일정 규칙으로 나열된 수열의 곱(i: 곱의 색인)

$$\prod_{i=1}^{n} a_i = a_1 \times a_2 \times \cdots \times a_{n-1} \times a_n$$

규칙

c는 상수, x_i와 y_i는 실수, k와 n은 양의 정수일 때 규칙은 다음과 같다.

(1) $\prod_{i=1}^{n} c = c^n$

증명 $$\prod_{i=1}^{n} c = \underbrace{c \times c \times \cdots \times c \times c}_{n\text{개}} = c^n$$

2. 수의 연산-곱 연산(Π와 !)

(2) $\prod_{i=1}^{n} x_i y_i = \prod_{i=1}^{n} x_i \times \prod_{i=1}^{n} y_i$

증명 $\prod_{i=1}^{n} x_i y_i = x_1 y_1 \times x_2 y_2 \times \cdots \times x_n y_n = (x_1 \times x_2 \cdots \times x_n) \times (y_1 \times y_2 \times \cdots \times y_n)$

$$= \prod_{i=1}^{n} x_i \times \prod_{i=1}^{n} y_i$$

2. 수의 연산-곱 연산(Π와 !)

(3) $\prod_{i=1}^{n} x_i = \prod_{i=1}^{k} x_i \times \prod_{i=k+1}^{n} x_i$ (단, $1 \leq k < n$)

증명 $$\prod_{i=1}^{n} x_i = x_1 \times x_2 \times \cdots \times x_k \times x_{k+1} \times \cdots \times x_{n-1} \times x_n$$
$$= (x_1 \times x_2 \times \cdots \times x_k) \times (x_{k+1} \times \cdots \times x_{n-1} \times x_n)$$
$$= \prod_{i=1}^{k} x_i \times \prod_{i=k+1}^{n} x_i$$

- 예) 1부터 20 사이에 있는 홀수의 곱

2. 수의 연산-곱 연산(Π와 !)

예제 2-8

다음을 연산하라.

(1) $\prod_{i=8}^{15} 3$ (2) $\prod_{i=0}^{4} (10-2i)$

2. 수의 연산-곱 연산(Π와 !)

예제 2-8

(3) $\prod_{k=1}^{5} 2i$

(4) $\prod_{i=1}^{4} (i - i^2)$

2. 수의 연산-곱 연산(Π와 !)

정의 2-9 계승(Factorial: $n!$)

1부터 시작하여 n까지 1씩 증가하는 수열의 곱. (단, $0! = 1$)

$$n! = 1 \times 2 \times 3 \times \cdots \times n = \prod_{i=1}^{n} i$$

예제 2-9

다음을 연산하라.

(1) $5!$

(2) $13!$

2. 수의 연산- 나누기 연산(|)

정의 2-10 나누기 연산: $d \mid n$

정수 n을 0이 아닌 정수 d로 나누어 몫 q를 구하는 연산($d \neq 0$)

$n = dq$를 만족하는 정수 q를 구하는 연산

$d \mid n$: d로 n을 나눌 수 있다.

$d \nmid n$: d로 n을 나눌 수 없다.

- q: 몫
- d: n의 약수Divisor 또는 인수Factor
- n: d의 배수

규칙

a, b, c, d, m, n은 정수

(1) $d \mid m$이고 $d \mid n$이면 $d \mid (m+n)$

증명 $d \mid m$에서 d로 m을 나눌 수 있으므로 $m = dk(k \in Z)$이고, $d \mid n$에서 d로 n을 나눌 수 있으므로 $n = dl(l \in Z)$이 된다. 그러므로 $m+n = dk+dl = d(k+l)$이 된다. 따라서 d로 $m+n$을 나눌 수 있다.

$$\therefore\ d \mid (m+n)$$

2. 수의 연산- 나누기 연산(|)

(2) $d \mid m$이고 $d \mid n$이면 $d \mid (m-n)$

증명 $d \mid m$에서 d로 m을 나눌 수 있으므로 $m = dk(k \in Z)$이고, $d \mid n$에서 d로 n을 나눌 수 있으므로 $n = dl(l \in Z)$이 된다. 그러므로 $m - n = dk - dl = d(k-l)$이 된다. 따라서 d는 $m-n$을 나눈다.

$$\therefore\ d \mid (m-n)$$

(3) $d \mid m$이면, $d \mid nm$

증명 $d \mid m$에서 d로 m을 나눌 수 있으므로 $m = dk(k \in Z)$이고, 양변에 n을 곱하면 $nm = dkn$이 된다. 따라서 d로 nm을 나눌 수 있다.

$$\therefore\ d \mid nm$$

2. 수의 연산- 나누기 연산(|)

(4) $a \mid b$이고 $b \mid c$이면 $a \mid c$

증명 $a \mid b$에서 a로 b를 나눌 수 있으므로 $b = ak(k \in Z)$가 되고, $b \mid c$에서 b로 c를 나눌 수 있으므로 $c = bl(l \in Z)$이 된다. 그러므로 $c = (ak)l$이 된다. 따라서 a로 c를 나눌 수 있다.

$$\therefore\ a \mid c$$

2. 수의 연산- 나누기 연산(|)

예제 2-10

다음 중 맞는 표현을 고르고, 나누어떨어지면 몫을 구하라.

(1) $3 \mid 9$　　(2) $7 \nmid 42$　　(3) $8 \mid 10$　　(4) $6 \nmid 15$　　(5) $10 \mid 100$

2. 수의 연산- 나머지 연산(mod)

정의 2-11 나머지 연산: $n \bmod d$

정수 n을 d로 나누어 몫이 q이고 나머지가 r일 때, r을 구하는 연산

$n = dq + r$을 만족하는 정수 r을 구하는 연산

$$n \bmod d = r$$

- q: 몫
- d: n을 나누는 수
- r: 나머지, $0 \leq r < d$

나머지 연산에서 나머지가 0일 경우, 나누기 연산(|)과 같으므로   다음이   성립

"$n \bmod d = 0$이면 $d \mid n$이고, $d \mid n$이면 $n \bmod d = 0$이다."

2. 수의 연산- 나머지 연산(mod)

예제 2-11

다음을 연산하여 나누기 연산으로 나타낼 수 있는 연산은 나누기 연산으로 표현하라.

(1) $27 \bmod 4$ (2) $52 \bmod 4$ (3) $7 \bmod 3$

(4) $25 \bmod 5$ (5) $-29 \bmod 4$

3 . 수의 표현

정의 2-12 n진법 / n진수

0과 $n-1$ 사이의 숫자들을 이용해 수를 표현하는 방식 또는 그렇게 표현된 수

- 현재 n 진법을 사용하고 있다면 n 을 기수Base Number
- 기수표기
  - 기수는 표현된 수의 오른쪽 하단에 표기

❖ 진법별 표현 – 10진수

정의 2-13 10진수(Decimal Number)

기수를 10으로 하는 수 체계, 0과 9 사이의 숫자를 이용해 수를 표현

- 정수 n에 대해($k > 0$, $0 \le a \le 9$)

$$n_{10} = a_k a_{k-1} \cdots a_1 a_0 = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10^1 + a_0 10^0$$

- 실수 n에 대해($k, l > 0$, $0 \le a \le 9$)

$$\begin{aligned} n_{10} &= a_k a_{k-1} \cdots a_1 a_0 \cdot a_{-1} a_{-2} \cdots a_{-l} a_{-(l+1)} \cdots \\ &= a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10^1 + a_0 10^0 \\ &\quad + a_{-1} 10^{-1} + a_{-2} 10^{-2} + \cdots + a_{-l} 10^{-l} + a_{-(l+1)} 10^{-(l+1)} + \cdots \end{aligned}$$

예제 2-12

다음 10진수를 기수와 자릿수를 이용해 풀어써라.

(1) 1582_{10}

(2) 523.6218_{10}

- 진법별 표현 – 2진수

정의 2-14 2진수(Binary Number)

기수를 2로 하는 수 체계, 0과 1을 이용해 수를 표현

- 정수 n에 대해($k > 0$, $a = 1$ or 0)

$$n_2 = a_k a_{k-1} \cdots a_1 a_0 = a_k 2^k + a_{k-1} 2^{k-1} + \cdots + a_1 2^1 + a_0 2^0$$

- 실수 n에 대해($k, l > 0$, $a = 1$ or 0)

$$\begin{aligned} n_2 &= a_k a_{k-1} \cdots a_1 a_0 \cdot a_{-1} a_{-2} \cdots a_{-l} a_{-(l+1)} \cdots \\ &= a_k 2^k + a_{k-1} 2^{k-1} + \cdots + a_1 2^1 + a_0 2^0 \\ &\quad + a_{-1} 2^{-1} + a_{-2} 2^{-2} + \cdots + a_{-l} 2^{-l} + a_{-(l+1)} 2^{-(l+1)} + \cdots \end{aligned}$$

예제 2-13

다음 2진수를 기수와 자릿수를 이용해 풀어써라.

(1) 1101001_2

(2) 10001.001101_2

• 진법별 표현 – 8진수

정의 2-15 8진수(Octal Number)

기수를 8로 하는 수 체계, 0과 7 사이의 숫자를 이용해 수를 표현

- 정수 n에 대해($k > 0,\ \ 0 \le a \le 7$)

$$n_8 = a_k a_{k-1} \cdots a_1 a_0 = a_k 8^k + a_{k-1} 8^{k-1} + \cdots + a_1 8^1 + a_0 8^0$$

- 실수 n에 대해($k, l > 0,\ 0 \le a \le 7$)

$$\begin{aligned} n_8 &= a_k a_{k-1} \cdots a_1 a_0 \cdot a_{-1} a_{-2} \cdots a_{-l} a_{-(l+1)} \cdots \\ &= a_k 8^k + a_{k-1} 8^{k-1} + \cdots + a_1 8^1 + a_0 8^0 + a_{-1} 8^{-1} + a_{-2} 8^{-2} + \cdots \\ &\quad + a_{-l} 8^{-l} + a_{-(l+1)} 8^{-(l+1)} + \cdots \end{aligned}$$

예제 2-14

다음 8진수를 기수와 자릿수를 이용해 풀어써라.

(1) 6351_8 (2) 712.3654_8

- 진법별 표현 – 16진수

정의 2-16 16진수(Hexadecimal Number)

기수를 16으로 하는 수 체계, 0~9와 A(10)와 F(15) 사이의 숫자를 이용해 수를 표현

- 정수 n에 대해($k > 0,\ 0 \le a \le 9$ 또는 $\mathrm{A} \le a \le \mathrm{F}$)

$$n_{16} = a_k a_{k-1} \cdots a_1 a_0 = a_k 16^k + a_{k-1} 16^{k-1} + \cdots + a_1 16^1 + a_0 16^0$$

- 실수 n에 대해($k, l > 0,\ 0 \le a \le 9$ 또는 $\mathrm{A} \le a \le \mathrm{F}$)

$$\begin{aligned} n_{16} &= a_k a_{k-1} \cdots a_1 a_0 \cdot a_{-1} a_{-2} \cdots a_{-l} a_{-(l+1)} \cdots \\ &= a_k 16^k + a_{k-1} 16^{k-1} + \cdots + a_1 16^1 + a_0 16^0 \\ &\quad + a_{-1} 16^{-1} + a_{-2} 16^{-2} + \cdots + a_{-l} 16^{-l} + a_{-(l+1)} 16^{-(l+1)} + \cdots \end{aligned}$$

- 진법별 표현 – 16진수

예제 2-15

다음 16진수를 기수와 자릿수를 이용해 풀어써라.

(1) 6921_{16} (2) $A41C_{16}$ (3) $9B.FE3_{16}$

진법별 사칙연산

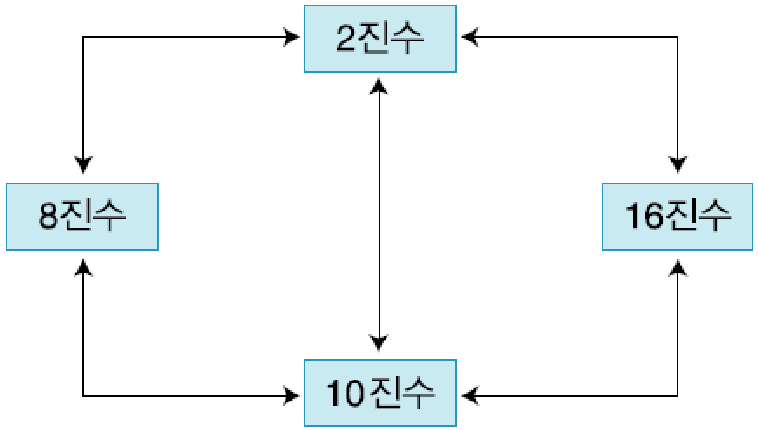
- 사칙연산의 기본 원리는 진법에 상관없이 같음
- 뺄셈에서 빌림수가 다를 뿐임
  - 10진수에서는 위의 자리에서 10을 빌려
  - 2진수에서는 2를, 8진수에서는 8을
  - 16진수에서는 16을 빌려옴
- 덧셈에서는 수 체계에 상관없이 올림수가 1

예제 2-16

다음 2진수를 연산하라.

(1) $101_2 + 11_2$　　　　(2) $100_2 - 11_2$

❖ 진법 간의 변환



[그림 2-2] 진법 간의 변환

- 10진수 → 2진수/8진수/16진수
  - 정수부 : 변환하려는 기수로 몫이 0이 될 때까지 나누면서 나오는 나머지를 나열
  - 소수부 : 소수부가 0이 될 때까지 변환하려는 기수로 곱함

- 진법 간의 변환

예제 2-19

다음 10진수를 2진수로 변환하라.

(1) 274_{10}

(2) 163.875_{10}

- 진법 간의 변환

예제 2-20

다음 10진수를 8진수로 변환하라.

(1) 274_{10}

(2) 163.875_{10}

• 진법 간의 변환

예제 2-21

다음 10진수를 16진수로 변환하라.

(1) 274_{10}

(2) 163.875_{10}

2진수/8진수/16진수 → 10진수

- 모든 진수는 기수와 수를 구성하는 숫자의 자릿수를 이용해 풀 수 있으므로 이를 이용해 10진수로 변환할 수 있다.

예제 2-22

다음을 10진수로 변환하라.

(1) 10101_2 (2) 1101.001_2 (3) 724_8

(4) 365.114_8 (5) $3CA_{16}$ (6) $E1.F01_{16}$

- 2진수와 8진수 간의 변환
  - 8진수는 0과 7 사이의 숫자로 수를 표현하는 진법
  - 8진수 한자리를 2진수로 표현하는 데는 3비트가 필요함
  - 2진수를 8진수로 변환할 때는 2진수를 소수점 기준으로 3비트씩 나누고 각 3비트 블록을 10진수로 변환

예제 2-23

2진수 11100101.0100111101_2을 8진수로 변환하라.

- 2진수와 8진수 간의 변환
  - 8진수를 2진수로 변환할 때는 8진수의 각 자리를 3비트의 2진수로 변환

예제 2-24

8진수 345.2364_8를 2진수로 변환하라.

- 2진수와 16진수 간의 변환
  - 16진수는 0~9와 10에서 15까지를 의미하는 영문 대문자 A와 F 사이의 숫자와 문자로 수를 표현하는 진법
  - 16진수 한자리를 2진수로 표현하는 데에 4비트가 필요
  - 2진수를 16진수로 변환할 때는 2진수를 소수점을 기준으로 4비트씩 나누고, 각 4비트 블록을 10진수로 변환
  - 16진수를 2진수로 변환할 때는 16진수의 각 자리를 4비트의 2진수로 변환

예제 2-25

2진수 11100101.0100111101_2을 16진수로 변환하라.

- 2진수와 16진수 간의 변환
  - 16진수를 2진수로 변환할 때는 16진수의 각 자리를 4비트의 2진수로 변환

예제 2-26

16진수 $E5.4F4_{16}$를 2진수로 변환하라.

예제 2-26

16진수 E5.4F4$_{16}$를 2진수로 변환하라.

4. 컴퓨터에서의 수의 표현과 연산

- 뺄셈과 나눗셈은 주어진 데이터 표현에 대한 보수를 사용하여 덧셈과 시프트 연산으로 연산
- 따라서 주어진 데이터에 대한 보수 변환의 이해가 필요

보수

- 보충해주는 수
- 어떤 수 a에 대한 n의 보수는 a와의 합이 n이 되는 수
- 10진수에서 1에 대한 보수는 9

4. 컴퓨터에서의 수의 표현과 연산

정의 2-17 2의 보수(2's Complement)

어떤 수 a와의 합이 $2(10_2)$가 되는 수

정의 2-18 1의 보수(1's Complement)

어떤 수 a와의 합이 $1(1_2)$이 되는 수

- 2진수를 구성하는 수는 0과 1뿐임
- 2진수를 구성하는 숫자들에 대한 1의 보수

  0에 대한 1의 보수: $0 + \chi = 1 \quad \therefore \chi = 1$

  1에 대한 1의 보수: $1 + \chi = 1 \quad \therefore \chi = 0$
- 이 수에 대한 2의 보수

  0에 대한 2의 보수: $0 + \chi = 0 \quad \therefore \chi = 0$

  1에 대한 2의 보수: $1 + \chi = (1)0 \quad \therefore \chi = 1$

4. 컴퓨터에서의 수의 표현과 연산

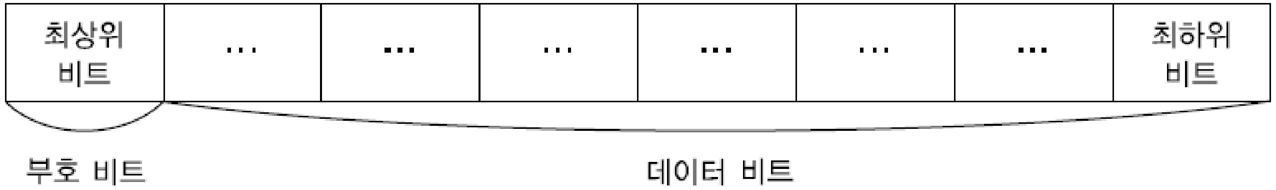
예제 2-27

다음 10진수에 대한 1의 보수와 2의 보수를 각각 구하라.

(1) 38_{10} (2) 107_{10} (3) 310_{10}

❖ 컴퓨터에서의 수의 표현

- 컴퓨터는 데이터를 워드Word 단위로 처리
- 워드의 맨 왼쪽에 위치하는 최상위 비트는 숫자 표현에서 부호를, 문자 등의 표현에서는 구분 기준으로 사용
- 데이터를 부호화-절댓값, 부호화-1의 보수, 부호화-2의 보수로 표현



정의 2-19 부호화-절댓값 표현

워드의 데이터 비트를 데이터의 절댓값으로 표현

예제 2-28

1워드가 8비트일 때, 10진수 $+53_{10}$과 -53_{10}을 부호화-절댓값으로 표현하라.

4. 컴퓨터에서의 수의 표현과 연산

정의 2-20 부호화-1의 보수 표현

워드의 데이터 비트를 1의 보수로 표현

(1) 음수에 대한 부호화−절댓값 표현에서 부호 비트는 그대로 사용한다.

(2) 음수에 대한 부호화−절댓값 표현에서 절대값 비트는 0은 1로, 1은 0으로 바꿔서 표현한다.

정의 2-21 부호화-2의 보수 표현

워드의 데이터 비트를 2의 보수로 표현

(1) 음수에 대한 부호화−절댓값 표현에서 부호 비트는 그대로 사용한다.

(2) 음수에 대한 부호화−절댓값 표현에서 절대값 비트에 대한 1의 보수에 1을 더한다.

4. 컴퓨터에서의 수의 표현과 연산

예제 2-29

1워드가 8비트일 때, 10진수 $+53_{10}$과 -53_{10}을 부호화-1의 보수와 부호화-2의 보수로 각각 표현하라.

4. 컴퓨터에서의 수의 표현과 연산

정리 2-2 데이터의 범위

1워드가 n비트인 컴퓨터에서

부호화-절댓값 표현: $-(2^{n-1}-1) \sim (2^{n-1}-1)$

부호화-1의 보수 표현: $-(2^{n-1}-1) \sim (2^{n-1}-1)$

부호화-2의 보수 표현: $-2^{n-1} \sim (2^{n-1}-1)$

4. 컴퓨터에서의 수의 표현과 연산

❖ 부호화-절댓값

- 컴퓨터에서 값을 표현하고 연산하는 데 한계가 있음
- 연산의 결과가 정확하지 않고 0을 표현하는 방법이 두 가지이기 때문
- $-3_{10} - 4_{10} = (-3_{10}) + (-4_{10})$ 을 연산 해보기

- 초과(Overflow)
  - 입력이나 연산의 결과가 1워드를 넘는 경우로, 1워드가 n비트일 때 n+1비트가 되는 경우
  - 이때 읽히는 것은 데이터 표현의 하위 1워드임

- 0_{10}을 부호화-절댓값으로 표현 : 0000($+0_{10}$) , 1000(-0_{10})
  - 연산에서 0의 두 가지 표현에 의해 연산이 복잡
  - 0은 중립 원소인데 표현이 통일되지 못한다는 한계 발생

4. 컴퓨터에서의 수의 표현과 연산

- 부호화-1의 보수
  - $-3_{10}-4_{10}=(-3_{10})+(-4_{10})$ 을 연산 해보기
    부호화-1의 보수 표현은 각각 1100, 1011
    1의 보수 표현으로 계산한 값은 결과가 맞음(7_{10})

  - 0_{10}을 부호화-1의 보수로 표현 : 0000(+0_{10}) , 1111(-0_{10})
  - 부호화-절댓값처럼 연산과정에서 0의 두 가지 표현을 고려해 야 하므로 연산이 복잡해지고, 0은 중립 원소인데 표현이 통일되지 못한다는 한계가 여전히 존재

4. 컴퓨터에서의 수의 표현과 연산

- 부호화-2의 보수 표현
  - $-3_{10} - 4_{10} = (-3_{10}) + (-4_{10})$ 을 연산 해보기

  - 0_{10}을 부호화-2의 보수로 표현하면 $+0_{10}$ 그대로 0000
    -0_{10}은 10000으로 초과로 발생하는 첫번째 비트의 1을 무시하면 0000
    $+0_{10}$ 과 -0_{10} 이 통일, 연산도 0000만 생각하므로 복잡하지 않음
    그래서 컴퓨터에서는 부호화-2의 보수 표현을 이용해 연산함

4. 컴퓨터에서의 수의 표현과 연산

❖ 보수의 10진수 변환

- 1워드가 n비트일 때, 부호화-1의 보수의 값을 10진수로 변환하는 방법
  - 주어진 1의 보수를 다시 1의 보수로 변환한 후, 이를 10진수로 변환(이때 부호는 -)

예제 2-30

1워드가 8비트일 때, 부호화-1의 보수로 표현된 11001011을 앞서 소개한 두 가지 방법으로 10진수로 변환하라.

4. 컴퓨터에서의 수의 표현과 연산

- 부호화-2의 보수 표현의 10진수로 변환 하는 방법
  - 주어진 2의 보수를 다시 2의 보수로 변환한 후, 이를 10진수로 변환

예제 2-31

1워드가 8비트일 때, 부호화−2의 보수로 표현된 11001100을 앞서 소개한 두 가지 방법으로 10진수로 변환하라.

4. 컴퓨터에서의 수의 표현과 연산

보수의 연산

- 보수 연산의 결과 발생한 초과비트(오버플로우)
  - 부호화-1의 보수 : 초과 비트는 한 번 더 더해줌
  - 부호화-2의 보수 : 초과 비트는 버림
- 초과비트 처리 후 부호비트 값이
  - 0 이면 그대로 10진수 변환
  - 1 이면 해당 보수로 다시 변환하고 10진 변환후 음수기호

예제 2-32

1워드가 8비트일 때, $13_{10} - 72_{10}$를 부호화 -1의 보수와 부호화 -2의 보수를 이용해 각각 연산하라.

4. 컴퓨터에서의 수의 표현과 연산

- 부호화-1의 보수로 연산 : 초과가 발생하면 초과된 비트를 최하위 비트에 더해줌

예제 2-33

1워드가 8비트일 때, 다음을 부호화−1의 보수로 연산하라.

(1) $33_{10} - 15_{10}$

(2) $-38_{10} - 70_{10}$

4. 컴퓨터에서의 수의 표현과 연산

- 부호화-2의 보수 표현으로 연산했을 때 초과가 발생하면 무시

예제 2-34

1워드가 8비트일 때, 다음을 부호화−2의 보수로 연산하라.

(1) $33_{10} - 15_{10}$

(2) $-38_{10} - 70_{10}$