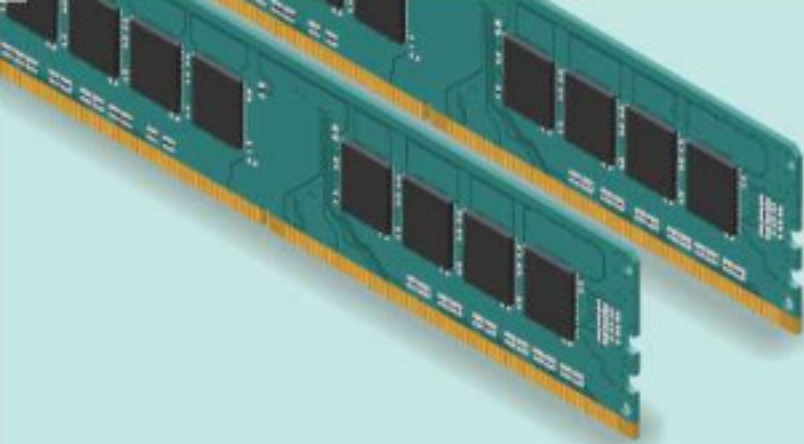




<http://blog.naver.com/sbwomen11/220555788773>

현대 사회에서 사람들은 빅 데이터를 통해 시장을 분석하고, 소비자들의 소비 형태를 이해할 수 있으며, 이러한 빅 데이터 활용으로 미래의 산업 전망까지도 내다볼 수 있다. 빅 데이터를 원하는 형태의 정보로 가공하기 위해서는 컴퓨터 시스템을 많이 사용하며, 가공된 정보는 필요에 따라 수치, 문자, 소리, 그림, 동영상으로 표현된다.

이 단원에서는 컴퓨터 시스템에서 사용하는 수의 체계를 알아보고, 이진수의 연산 과정과 실생활에서 접하는 수치와 문자, 소리, 그림, 동영상 등 다양한 정보들이 어떻게 표현되는지 알아본다.

I

정보의 표현

> CHAPTER <

- 01. 수의 체계
- 02. 디지털 정보의 연산
- 03. 디지털 정보의 표현



수의 체계

학 습 목 표

- 자료와 정보에 대하여 설명할 수 있다.
- 컴퓨터 시스템에서 사용하는 수의 체계와 여러 진법으로 표현되는 원리를 설명할 수 있다.
- 진법 변환 방법을 설명할 수 있다.
- 다른 진수로 변환할 수 있다.

여는 이야기

우리 주위에서 사용하는 다양한 진수들



EBS 다큐 프라임'과학 대기화 - 다섯 개의 열쇠 1부 : 0과 1, 디지털'2011. 9. 12.

모든 정보를 표현할 수 있는 가장 간단한 기호 0과 1, 우리 인류의 기본 정보를 외계에 있을 생명체에게 보내는 정보도 10진법이 아닌 2진법으로 전파를 만들어 보내고 있다. 그것이 아레시보 메시지인데 가장 간단한 기호 체계이므로 외계 생명체가 존재한다면 1~10까지 숫자, 원자번호, DNA 이중나선 구조 등의 정보를 해석할 수 있다는 근거에서 2진법으로 만든 것이다.

단/원/학/습/안/내

이 단원에서는 컴퓨터 시스템에서 사용하는 수의 체계에 대하여 알아보고 우리 생활에서 주로 사용하는 10진법과 컴퓨터 시스템에서 사용하는 2진법의 표현 원리에 대하여 살펴본다. 또한, 서로 다른 진법으로 변환하는 방법을 익혀 컴퓨터 시스템 또는 디지털 정보 가공과 관련된 직무를 수행하는 데 필요한 실무 능력을 기르도록 한다.

우리는 일상생활에서 컴퓨터 시스템을 통해 다양한 정보를 접하며, 이를 활용하고 있다. 컴퓨터 시스템은 자료(Data)를 입력받아 다양한 처리 과정을 거쳐 사람의 의사 결정에 도움을 주는 정보를 생산한다.

(1) 자료와 정보의 개념

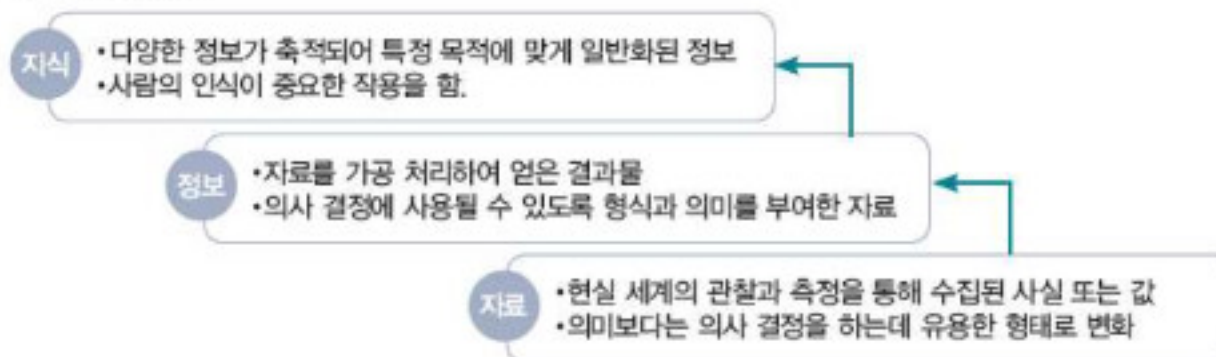
버스나 지하철과 같이 대중교통을 이용할 때는 버스 노선이나 지하철 노선과 시간이나와 있는 앱을 이용한다. 또한 외출할 경우에는 날씨를 확인하고 날씨에 맞는 적절한 옷차림과 우산, 양산 등을 준비한다. 뿐만 아니라 물건을 구입할 경우에는 인터넷에서 미리 가격, 품질 등을 조사하고 구입하는 경우가 많다. 이처럼 우리는 일상생활에서 많은 정보를 접하고 있으며, 이를 적절히 활용한다. 그런데 이러한 정보는 여러 자료들을 가공하여 만들어지게 되며 이렇게 만들어진 정보는 축적되어 지식으로 일반화된다.

예를 들어 일기 예보는 작성된 일기도를 기준으로 발표하는데, 일기도는 여러 지역에서 측정하고 조사한 기압, 온도, 풍속 등을 토대로 만들어진다. 이때 기압, 온도, 풍속이 자료에 해당되고 일기도는 정보에 해당된다.

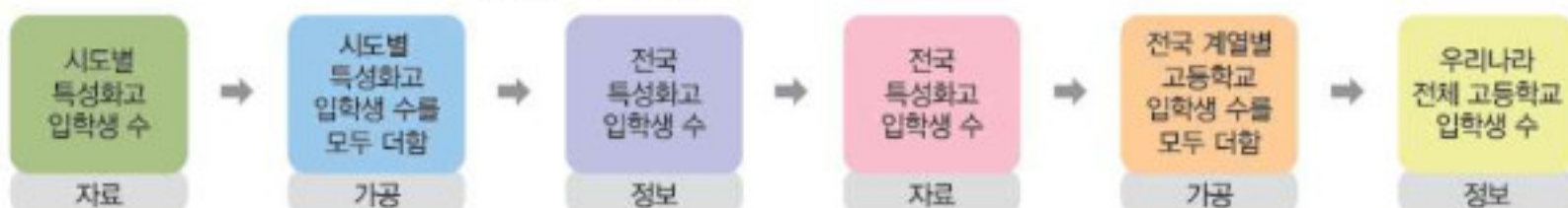
이와 같이 자료는 관찰이나 측정을 통해 수집한 단순한 사실이나 값들의 집합을 의미하며, 정보는 수집된 자료를 다양한 방법으로 가공, 처리하여 사용자가 의미 있게 사용할 수 있도록 생성된 자료이다. 온도, 습도, 비의 양 등은 하나의 자료이지만, 여행을 가는 사람에게는 정보가 될 수 있다. 그리고 정보는 또 다른 정보를 만들기 위한 자료가 될 수도 있다.

자료와 정보

- 자료(data)는 관찰이나 측정을 통해 수집한 단순한 사실이나 값
- 정보(information)는 수집된 자료를 가공, 처리하여 사용자가 실제 생활에 도움이 될 수 있도록 만든 것



[자료, 정보, 지식의 관계]



전국특성화고 입학생수는 시도별 특성화고 입학생 수를 이용하여 나온 정보이지만 우리나라 고등학교의 입학생 수를 구하는 자료이기도 하다.

[자료와 정보의 예]

그림 1-1 자료, 정보, 지식의 관계와 예

(2) 정보 처리 과정

정보의 특성

- 정보는 필요한 시점에 제공될 때 그 가치가 있다. 교통 정보는 출·퇴근 시간과 같이 사용자가 원하는 특정한 시점에서만 그 정보가 가치를 갖는다.
- 정보는 많이 사용해도 없어지지 않으며, 많은 사람이 공유할수록 더 큰 가치를 가진다. 지진이나 태풍과 같은 재난·재해 정보는 많은 사람이 공유할 때 더 큰 가치를 갖는다.
- 같은 정보라도 사용자의 필요에 따라 그 가치가 달라진다. 영화에 관심 있는 사람에게는 현재 상영 중인 영화에 대한 정보가 그렇지 않은 다른 사람에 비해 더 큰 가치를 갖는다.
- 정보는 활용할 경우에만 정보로서의 가치를 발휘한다. 아무리 좋은 정보도 활용하지 않으면 정보로서의 의미가 없어진다. 일기 예보에 비가 온다는 정보를 듣고도 외출 시 우산을 준비하지 않으면 정보로서의 가치가 없어진다.

자료가 정보로 만들어지기 위해서는 일상생활이나 자연 현상에서 발생된 많은 자료를 수집하여 이용이 가능한 형태로 가공해야 하는데, 이러한 과정을 자료 처리(Data Processing) 또는 정보 처리(Information Processing)라고 한다. 자료를 처리할 때 자료의 양이 많고 복잡한 과정일 경우에는 이를 자동으로 처리할 수 있는 컴퓨터 시스템이 필요하다. 다음 그림 1-2는 자료가 처리 과정을 거쳐 정보가 되는 과정을 나타낸 그림이다.



그림 1-2 정보 처리 과정

(3) 컴퓨터 시스템에서 자료 표현

우리가 일상생활에서 사용하는 수와 달리 컴퓨터 시스템 내부에서 다루는 자료는 전기에 의해 동작하므로 '전기가 흐른다'와 '전기가 흐르지 않는다'의 두 가지 신호만을 인식하는 1과 0의 두 가지 상태로 자료를 기억하고 처리한다. 0과 1의 두 가지 수를 표현할 수 있는 2진수 한자리를 비트(bit)라 하며, 8비트의 묶음을 바이트(byte)라 한다.



더 알아보기

컴퓨터 시스템에서 사용하는 자료의 단위

비트(bit) : 컴퓨터 시스템에서 사용하는 가장 작은 단위로 0과 1로 이루어짐

바이트(byte) : 8개의 비트로 이루어짐. 컴퓨터 시스템에서 문자를 다루는 최소 단위

킬로바이트(KB) : 1024개의 바이트로 이루어짐. K(킬로)는 원래 '10을 세번 곱한 것인 1000을 뜻하지만, 컴퓨터의 단위로 쓸 때는 1024로 쓰이며, 1024는 2를 10번 곱한 값임($2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^{10}$).

메가바이트(MB) : 1024개의 킬로바이트로 이루어짐. K(킬로)가 1024개 모인 것을 뜻함

기가바이트(GB) - 테라바이트(TB) - 페타바이트(PB) - 엑사바이트(EB)

2 진법

진법은 수를 표시하는 방법 중의 하나로, 숫자의 위치에 따라 가중치가 달라지는 수로 나타내는 방법이다. 수는 우리가 일상생활에서 사용하는 10진법뿐만 아니라 2진법, 8진법과 16진법 등 다양한 방법으로 표현할 수 있다.

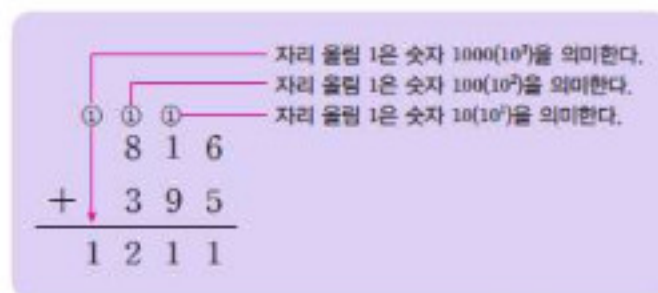
가중치

표현된 숫자의 위치가 갖는 자릿값이다. 십진수 763에서 7은 100, 6은 10, 3은 1의 가중치를 갖는다.

(1) 10진법

10진법은 0부터 9까지 10개의 숫자를 사용하여 수를 표현하는 방법을 말하며, 10진법에 의해 표현된 수를 10진수라고 한다.

10진법에서는 각 자리에 위치한 값이 9를 초과하게 되면 왼쪽으로의 1의 자리 올림이 발생한다. 이때 자리 올림 1은 오른쪽에 위치한 값보다 10배의 가중치를 가진다.



진수 표현

수의 진법은 밑(base)으로 표현한다. 2진수의 밑은 2이고, 10진수의 밑은 10이다. 일반적으로 10진수의 밑은 생략한다.

$101_{(10)} \rightarrow$ 밑(base)

예제

10진수 375를 10진법의 전개식으로 표현해 보자.

풀이 $375 = 3 \times 10^2 + 7 \times 10^1 + 5 \times 10^0$
 $= 3 \times 100 + 7 \times 10 + 5 \times 1 = 300 + 70 + 5$
 즉, 375는 100이 3개, 10이 7개, 1이 5개인 수를 합친 수이다.

(2) 2진법

2진법은 0과 1의 2개의 숫자를 사용하여 수를 표현하는 방법을 말하며, 2진법에 의해 표현된 수를 2진수라고 한다. 2진법에서는 자리가 하나씩 올라감에 따라 자리의 값이 2배씩 커진다.

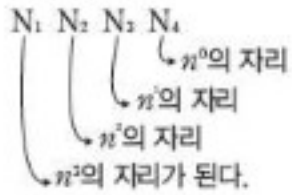


2진수의 가중치

$1101_{(2)}$ 에서 각 자릿수는 오른쪽부터 왼쪽 방향으로 2^0 , 2^1 , 2^2 , 2^3 의 가중치를 갖는다.

(3) 8진법과 16진법

n 진법의 수가 다음과 같을 때 자릿수의 구성



8진법은 0부터 7까지 8개의 숫자를 사용하여 수를 표현하는 방법을 말하며, 8진법에 의해 표현된 수를 8진수라고 한다. 그리고 16진법은 0부터 15까지 16개의 숫자를 사용하여 수를 표현하는 방법을 말하며, 16진법에 의해 표현된 수를 16진수라고 한다.

16진수는 16개의 숫자 중에서 10개는 10진수의 0부터 9를 그대로 사용하고, 나머지 6개인 10부터 15까지는 다른 진법의 수와 구별하기 위하여 10은 A, 11은 B, 12는 C, 13은 D, 14는 E, 15는 F로 각각 표현한다.

(4) 2진법과 컴퓨터

표 1-1 각 진법 사이의 관계

10진법	2진법	8진법	16진법	10진법	2진법	8진법	16진법
0	0000	00	0	10	1010	12	A
1	0001	01	1	11	1011	13	B
2	0010	02	2	12	1100	14	C
3	0011	03	3	13	1101	15	D
4	0100	04	4	14	1110	16	E
5	0101	05	5	15	1111	17	F
6	0110	06	6	16	10000	20	10
7	0111	07	7	17	10001	21	11
8	1000	10	8	18	10010	22	12
9	1001	11	9	19	10011	23	13

우리는 일상생활에서 10진법을 사용하고 있지만 컴퓨터 시스템 내부는 전등이 전기의 흐름에 따라 켜지고 꺼지는 두 가지 상태로 동작하는 것과 같이 모든 자료들을 0과 1의 두 가지 상태로 표현한다. 즉, 컴퓨터에서 연산을 수행하기 위해서는 입력한 10진수를 2진수로 변환하여 컴퓨터 내부에 전달하여 처리한 후, 이것을 출력하기 위해서는 2진수를 10진수로 변환하는 과정을 거쳐야 한다.



읽을거리

원의 각도가 왜 360도인가?

원은 360도이다. 그럼 누가 맨 처음 원은 360도라고 정했을까? 지금으로부터 약 4000년 전 지금의 중동 지역인 이라크, 시리아, 이스라엘 등의 나라가 자리잡고 있는 땅에 수준이 높은 문명 국가인 바빌로니아라는 나라가 있었다. 이 바빌로니아는 수의 진법, 각도, 측량 기술 등 수학과 관련된 연구가 활발했다. 그 당시에 만들어진 것이 1주일(7일), 시간의 12진법, 60진법(1시간이 60분, 1분이 60초) 등은 오늘날에도 쓰이고 있는 것들이다. 이 바빌로니아 사람들은 1년을 12달로 보고, 한 달을 30일로 나누어 날짜를 계산하였다. 그리고 그 시간의 오차는 '윤년'이라는 해를 두어, 그 해는 1년을 13달로 계산하였다. (이 방법은 우리 나라 윤년과 비슷하다.) 그런데 이때의 사람들은 1년을 원으로 나타내었고, 1년을 360일이라 하였다. 여기서 원의 각도도 360도가 되었다.

3

진법 변환

컴퓨터에 10진수 127을 입력하면 컴퓨터는 127을 2진수로 변환하여 입력하고 이를 처리한 후 다시 10진수로 변환하여 출력한다. 이와 같이 임의의 진법으로 표현된 자료를 다른 진법으로 바꾸는 것을 진법 변환이라고 한다.

(1) 2진수, 16진수에서 10진수로 변환

2진수, 8진수, 16진수로 표현된 수를 10진수로 변환할 때에는 변환하는 수의 밑수와 각 자리에 해당하는 가중치를 곱하여 이들을 더하면 된다. 이때, 2진수의 밑수는 2, 8진수의 밑수는 8, 16진수의 밑수는 16이 된다.

진법 변환

컴퓨터 내에서 자료 표현이나 연산을 위해서는 각 진법 사이의 변환이 어떻게 이루어지는지 알아야 한다.



예제

2진수 $10101_{(2)}$ 과 16진수 $AD4_{(16)}$ 를 10진수로 변환해 보자.

$$\begin{aligned} 10101_{(2)} &= 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 1 \times 16 + 0 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1 \\ &= 16 + 0 + 4 + 0 + 1 \\ &= 21_{(10)} \end{aligned}$$

$$\begin{aligned} AD4_{(16)} &= 10 \times 16^2 + 13 \times 16^1 + 4 \times 16^0 \\ &= 10 \times 256 + 13 \times 16 + 4 \times 1 \\ &= 2560 + 208 + 4 \\ &= 2772_{(10)} \end{aligned}$$

(2) 10진수에서 2진수, 16진수로 변환

10진수를 2진수나 16진수로 변환할 때는 정수 부분과 소수부분으로 나누어 각각 처리한다.

1 정수 부분의 변환 방법

10진수의 정수를 변환할 때는 변환하는 진수의 밑수로 나누어 몫과 나머지를 구하여 그 몫이 0이 될 때까지 계속 나눈 후, 구해진 나머지를 역순으로 표현한다.

2 소수 부분의 변환 방법

10진수의 소수 부분을 변환할 때는 변환하고자 하는 진수의 밑수로 소수점 이하가 0이 될 때까지 곱한 후, 정수 부분을 취하여 자리올림수가 발생한 순서대로 표현한다. 이때, 나머지 소수 부분이 0이 되지 않은 경우에는 몇 번의 곱셈을 수행하여 근삿값을 구한다.

소수 부분의 변환

나머지 소수 부분이 0이 되지 않는 경우에는 몇 번의 곱셈을 수행하여 근삿값을 구한다.

예제

10진수 30.6875를 2진수로 변환해 보자.

풀이 정수 부분 변환

$$\begin{array}{r}
 2 \overline{) 30} \\
 2 \overline{) 15} \cdots 0 \\
 2 \overline{) 7} \cdots 1 \\
 2 \overline{) 3} \cdots 1 \\
 2 \overline{) 1} \cdots 1 \\
 \quad 0 \cdots 1
 \end{array}
 \Rightarrow 30_{(10)} = 11110_{(2)}$$

소수 부분 변환

$$\begin{array}{r}
 0.6875 \\
 \times 2 \\
 \hline
 \textcircled{1} 3750 \\
 \times 2 \\
 \hline
 \textcircled{0} 7500 \\
 \times 2 \\
 \hline
 \textcircled{1} 5000 \\
 \times 2 \\
 \hline
 \textcircled{0} 0000
 \end{array}
 \Rightarrow 0.6875_{(10)} = 0.1011_{(2)}$$

$$\therefore 30.6875_{(10)} = 11110.1011_{(2)}$$

(3) 2진수에서 8진수, 16진수로 변환

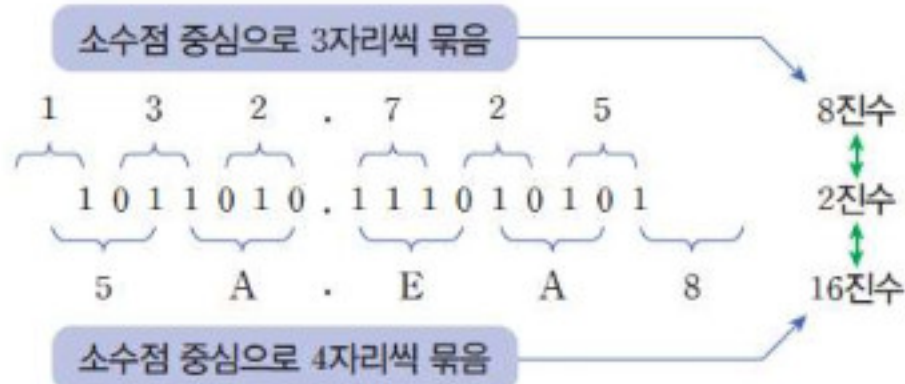
2진수에서 8진수, 16진수로 변환할 때는 소수점을 중심으로 각 자리를 묶어서 변환한다.

1 2진수에서 8진수로의 변환

2진수의 세 자리는 8진수의 한 자리로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 세 자리씩 묶어서 각 묶음에 대응되는 8진수의 한 자리로 표현한다.

2 2진수에서 16진수로의 변환

2진수의 네 자리는 16진수의 한 자리로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 네 자리씩 묶어서 각 묶음에 대응되는 16진수의 한 자리로 표현한다.



자리 묶음할 때 빈칸 처리

2진수에서 8진수와 16진수로 변환할 때 빈칸은 0으로 채워서 계산한다.

그림 1-3 2진수를 8진수로, 16진수로 변환하는 과정

직무 따라잡기

2진수 1101100.001101을 8진수와 16진수로 각각 변환해 보자.

풀이 8진수

--	--	--	--	--	--	--	--

$$1101100.001101_{(2)} = \underline{\hspace{2cm}}$$

16진수

--	--	--	--	--

$$1101100.001101_{(2)} = \underline{\hspace{2cm}}$$

16진수 6B7.D를 2진수로 변환해 보자.

풀이

(4) 8진수, 16진수에서 2진수로 변환

8진수, 16진수에서 2진수로 변환할 때는 소수점을 중심으로 8진수 한 자리는 2진수 세 자리로, 16진수 한 자리는 2진수 네 자리로 나타낸다.

1 8진수에서 2진수로의 변환

8진수의 한 자리는 세 자리의 2진수로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 8진수의 한 자리를 2진수의 세 자리로 표현한다.

예제

8진수 63,2를 2진수로 변환해 보자.

풀이 8진수 6 3 . 2
 $\underbrace{\quad\quad\quad}$ $\underbrace{\quad\quad\quad}$ $\underbrace{\quad\quad\quad}$
2진수 1 1 0 0 1 1 , 0 1 0
 $63,2_{(8)} = 110011,010_{(2)}$

2 16진수에서 2진수로의 변환

16진수의 한 자리는 네 자리의 2진수로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 16진수의 한 자리를 2진수의 네 자리로 표현한다.

예제

16진수 A5,B를 2진수로 변환해 보자.

풀이 16진수 A 5 . B
 $\underbrace{\quad\quad\quad\quad}$ $\underbrace{\quad\quad\quad\quad}$ $\underbrace{\quad\quad\quad\quad}$
2진수 1 0 1 0 0 1 0 1 , 1 0 1 1
 $A5,B_{(16)} = 10100101,1011_{(2)}$



활동 주제

2진수, 8진수, 10진수, 16진수를 각각 다른 진수로 변환하는 방법을 예를 정리하여 발표한다.

준비와 과정

- 모둠을 설정한다.
 - 4~5명의 모둠을 설정한다.
 - 모둠별로 모둠의 이름을 정하고, 모둠에서의 역할을 정한다.
- 진법 변환할 진수를 선택한다.
- 정수 부분과 소수 부분을 나누어 변환 방법을 알아본다.
- 정리한 자료를 발표자가 발표한다.
 - 다른 모둠이 발표할 때, 자신이 속한 모둠의 자료와 비교해 본다.

활동 내용

1. 변환할 진수를 찾아보자.

- | | |
|----------------------------|----------------------------|
| - 10진수를 2진수, 8진수, 16진수로 변환 | - 2진수, 8진수, 16진수를 10진수로 변환 |
| - 2진수를 8진수, 16진수로 변환 | - 8진수, 16진수를 2진수로 변환 |

2. 진법 변환 방법의 예

- ① 정수 부분의 변환 방법 : 변환하려고 하는 진법의 밑수로 나누고, 그 나머지를 구하여 뒤에서부터 취한다.
- 예를 들어 10진수 25를 2진수로 표현하려면 다음과 같이 구한다.

$$\begin{array}{r}
 2 \overline{) 25} \\
 2 \overline{) 12} \dots ① \\
 2 \overline{) 6} \dots ① \\
 2 \overline{) 3} \dots ① \\
 ① \dots ①
 \end{array}
 \quad \text{그러므로, } 11001_{(2)}$$

- 10진수 128을 8진수로 표현하려면 다음과 같이 구한다.

$$\begin{array}{r}
 8 \overline{) 128} \\
 8 \overline{) 16} \dots ① \\
 ② \dots ①
 \end{array}
 \quad \text{그러므로, } 200_{(8)}$$

- ② 소수 부분의 변환 방법 : 소수 부분의 수에 변환하고자 하는 진수의 밑수를 곱하여 자리 올림수가 발생하는 대로 차례로 적으며, 0이 될 때까지 반복한다.

$$\begin{array}{r}
 0,625 \\
 \times 2 \\
 \hline
 ①250 \\
 \times 2 \\
 \hline
 ①500 \\
 \times 2 \\
 \hline
 ①000
 \end{array}
 \quad \text{그러므로, } 0,101_{(2)}$$

평가 포인트 : 정수 부분과 소수 부분을 나누어 생각할 수 있도록 하고, 정수 부분은 구하려는 진수로 나누고 소수 부분은 구하려는 진수로 곱한다는 것을 알고 있는지, 또한 이때 구한 값에서 나온 정수 부분이 0인 경우를 제대로 표현할 수 있는지 확인한다.

학습목표

- 컴퓨터 시스템에서 처리하는 수치 데이터의 표현 원리를 설명할 수 있다.
- 컴퓨터 시스템에서 처리하는 2진수의 덧셈 연산 처리 원리를 설명할 수 있다.
- 컴퓨터 시스템에서 처리하는 2진수의 뺄셈 연산 처리 원리를 설명할 수 있다.
- 컴퓨터 시스템에서 처리하는 2진수의 곱셈과 나눗셈 연산 처리 원리를 설명할 수 있다.

여는 이야기

계산기의 시초들

파스칼은 톱니바퀴와 손잡이를 이용하여 계산하는 파스칼린이란 기계식 계산기를 만들었다. 파스칼린에는 여러 개의 톱니바퀴에 10개의 톱니가 있어 0에서 9까지의 순서대로 기록될 수 있게 8개의 톱니바퀴가 8단위의 수까지 나타낼 수 있게 되어 있다. 하나의 톱니바퀴에서 9에서 0으로 옮겨지면 그 옆에 있는 톱니바퀴에서 하나의 톱니가 움직일 수 있도록 하여 자동으로 다음 단위로 올라가게 만들었다.

파스칼린이 만들어진 지 30년 후에 독일의 라이프니치에 의해 더욱 개량되었다. 그는 새로운 계산기에 대한 관심으로, 곱셈과 나눗셈도 가능한 톱니바퀴식 계산기의 원리를 개발하였다. 그는 곱셈, 나눗셈을 각각 덧셈과 뺄셈으로 나타냈다. 예를 들어 32×3 은 $32+32+32$ 와 같이 생각했고, $45 \div 7$ 은 45에서 7을 계속 빼면 된다고 생각했다. 그러나 그는 파스칼의 계산기가 가지고 있던 덧셈과 뺄셈 기능에 완벽한 곱셈, 나눗셈 기능을 더했지만 실제 제작에는 관심을 가지지 않았다. 특히 라이프니치는 10개의 톱니바퀴를 쓰는 10진법 계산기가 아니라, 2진법의 응용에도 동시에 관심을 가지고 있었는데, 만약 그가 이진법을 응용하여 계산기를 만들어 냈다면 컴퓨터의 역사는 또 바뀌었을지도 모른다.



주판



파스칼린



라이프니치 계산기



에니악

단/원/학/습/안/내

이 단원에서는 컴퓨터 시스템에서 수치 데이터를 어떻게 표현하는지, 2진수의 덧셈, 뺄셈, 곱셈, 나눗셈은 어떤 원리로 처리하는지 알아봄으로써 컴퓨터 시스템의 원리에 관련된 직무를 수행하는 데 필요한 실무 능력을 향상시킬 수 있도록 한다.

1 수치 데이터의 표현

컴퓨터 시스템에서 연산을 위해 사용하는 수치 데이터는 2진수를 사용한다. 또한 10진수를 2진수로 변환하지 않고 각 자리를 2진수로 표현한 10진 데이터 형식이 있다. 그림 1-4는 컴퓨터 시스템에서 사용하는 수치 자료들이다.



그림 1-4 컴퓨터 시스템에서 사용하는 수치 데이터

(1) 정수 데이터 표현

수치 데이터를 2진수로 표현할 때 부호는 비트의 모임 중에서 가장 왼쪽에 위치한 최상위 비트 한 비트로 표현한다. 양수는 부호 비트를 0으로 표시하고 나머지 비트를 절댓값으로 표현한다. 음수는 부호화 절댓값 표현법, 1의 보수 표현법, 2의 보수 표현법이 있으며, 일반적으로 2의 보수 표현법이 널리 사용된다.

예제

양수 139와 음수 -10을 16비트로 표현해 보자.

풀이

(1) 양수 139 : 최상위 부호 비트를 0으로 표현하고, 나머지 비트를 절댓값으로 표현한다.

$\begin{array}{c} 0 \quad 000 \quad 0000 \quad 1000 \quad 1011 \\ \text{부호} \quad 0 \quad 0 \quad 8 \quad 8 \end{array} \rightarrow \begin{array}{l} \text{2진수의 표현} \\ \text{16진수의 표현} \end{array}$

(2) 음수 10의 표현

- ① 부호화 절댓값 표현법 → 부호만 '1'로 표현하고, 절댓값으로 표현한다.
- ② 1의 보수 표현법 → 양수 10의 전체 비트를 0을 1로, 1을 0으로 각각 반전시킨다.
- ③ 2의 보수 표현법 → 양수 10의 전체 비트를 0을 1로, 1을 0으로 각각 반전시킨 후, 끝자리에 1을 더한다.

$-10 \quad \boxed{1 \ 000 \ 0000 \ 0000 \ 1010}_2$
 $-10 \quad \boxed{1 \ 111 \ 1111 \ 1111 \ 0101}_2$
 $-10 \quad \boxed{1 \ 111 \ 1111 \ 1111 \ 0110}_2$

컴퓨터에서 정수를 표현할 때는 고정 소수점 데이터 형식을 사용한다. 고정 소수점 형식은 소수점의 위치가 그림 1-5와 같이 특정 위치에 고정되어 있다고 가정하고 정수를 표현하는 방식이다. 고정 소수점 데이터 형식은 2바이트와 4바이트 등으로 표현한다.

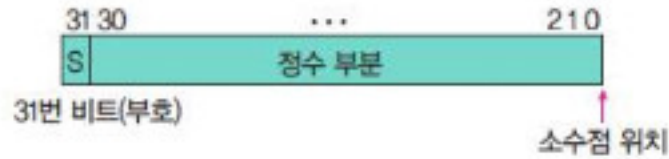


그림 1-5 4바이트 고정 소수점 데이터 형식

작업 따라잡기

10진수 23을 4바이트 고정 소수점 데이터 형식으로 표현해 보자.

(2) 실수 데이터의 표현

실수는 소수 부분을 포함하는 수이다. 여기서 소수는 0과 1사이의 수를 말한다. 2진수에서 실수를 표현하는 방법으로는 부동 소수점 데이터 형식을 사용한다.

부동 소수점 데이터 형식은 컴퓨터 내부에서 소수점이 있는 실수를 표현하는 방법이다. 부동 소수점 데이터 형식은 4바이트 단정도 실수형과 8바이트 배정도 실수형 등으로 표현한다. 그림 1-6은 10진수 213.45를 지수 형태로 표현한 것으로 그 값은 다음과 같다.

소수

0과 1사이의 수로
0.99999...9~0.0000...1

소수 표현의 예

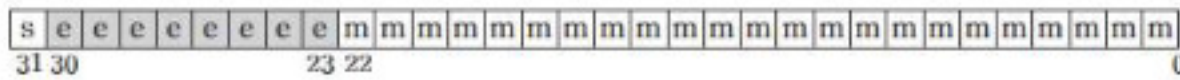
213.45_{10}
 101101.101_2

$$\begin{aligned} 213.45 &= 21345 \times 10^{-2} \\ &= 0.21345 \times 10^3 \end{aligned}$$

그림 1-6 10진수 213.45의 다양한 표현

부동 소수점 데이터 형식은 그림 1-7과 같이 부호는 부호부, 지수부, 가수부(소수부)로 구분한다. 부호부는 최상위 비트로, 가장 왼쪽에 있는 1비트를 말하며, 양수(+)이면 0으로, 음수(-)이면 1로 표현한다.

지수부는 지수를 2진수로 표현하고, 가수부는 소수점 아래 10진 유효 숫자를 2진수로 변환하여 표현한다. 이때 소수점은 지수부와 가수부 사이에 있는 것으로 가정한다.

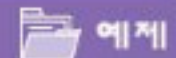


s = 부호
e = 바이어스된 지수
m = 정규화된 가수
부동 소수점 수는 $s \times 1, m \times 2^{(e-127)}$

그림 1-7 IEEE 754 표준에 따른 부동소수점 데이터 형식의 표현법

바이어스된 지수

지수부는 부호를 표현하지 않고 바이어스(bias)를 사용하여 지수의 양수와 음수를 표현한다. 즉 8비트로 지수부를 나타내면 수의 값은 0부터 255이므로 127을 기준으로 잡는다. 따라서 지수부가 127이면 0승, 128이면 1승, 129면 2승이 되고, 126이면 -1승, 125면 -2승을 나타낸다.



10진수 -123.25를 4바이트 부동 소수점 데이터 형식으로 표현해 보자.

- 풀이**
- 10진수를 2진수로 변환하기
 - 정수 부분: 123을 2진수로 표현하면 111 1011
 - 소수 부분: 0.25를 2진수로 표현하면 0.01
 - 10진수는 123.25은 2진수 1111011.010이다.
 - 정규화하기
 - 1111011.01은 1.11101101×2^6 이다.
 - 부호 비트, 지수부, 가수부 구하기
 - 부호 비트: 음수 이므로 1
 - 지수부: $127(\text{바이어스 값}) + 6 = 133$ 이를 2진수 8비트로 표현하면 1000 0101
 - 가수부: 정규화에서 정수 1를 제외한 값 11101101
 - 4바이트 부동 소수점 데이터 형식으로 표현하면
 - 1 1000 0101 1110 1101 0000 0000 0000 0000 (부호, 지수부, 가수부로 나누어 표현한 것)
 - 1100 0010 1111 0110 1000 0000 0000 0000 (실제 데이터에 저장된 형태를 4비트씩 나누어 표현한 것)

(3) 10진 데이터 형식

고정 소수점 데이터를 표현하는 방법 중의 하나로 10진수를 2진수로 변환하지 않고 10진수 상태로 표현하는 것이다. 이 형식의 종류에는 존형 10진 데이터 형식과 팩형 10진 데이터 형식 등이 있다.

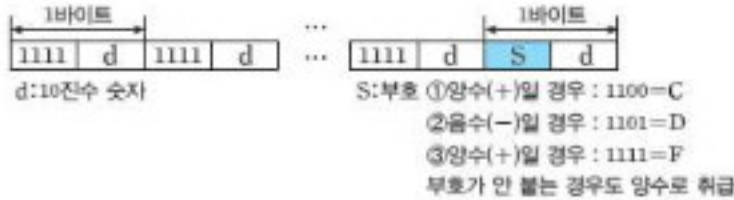
1 존형 10진 데이터 형식

10진수의 한 자리를 8개의 비트로 표현하는 방법으로, 8비트 중에서 왼쪽의 4비트는 존(zone), 나머지 4비트는 숫자(digit)를 나타낸다. 이때, 10진수의 부호는 가장 오른쪽 8비트 중 존 부분에 부호를 표현하는데, 존 부분이 양수(+)이면 C(1100)로, 음수(-)이면 D(1101)로 나타낸다.

● 여러 자리의 10진수를 표현하는 방법

- 10진수의 자릿수만큼 존 형식을 연결하여 사용
- 최하위 4비트의 존영역에 부호를 표시

양수(+): 1100 음수(-): 1101



[참고] 원래 부호가 없는 경우는 양수이므로 부호가 없이 1111로
존 부분이 표현되는 경우도 이를 양수로 취급하므로 양수의
부호는 1100과 1111 두 가지로 표현할 수 있다.

● 존 형식으로 10진수를 표현하는 예

• +213

1111	0010	1111	0001	1100	0011
F	2	F	1	C(+)	3
2		1		3	

• -213

1111	0010	1111	0001	1101	0011
F	2	F	1	D(-)	3
2		1		3	

[참고] 10진수는 자릿수만큼 연속된 바이트에 표현한다.
따라서 위의 213은 3바이트에 표현한다.

2 팩형 10진 데이터 형식

10진수의 한 자리를 4개의 비트로 표현하는 방법으로, 가장 오른쪽 4개의 비트는 부호를 표현하는데, 양수(+)이면 C(1100)로, 음수(-)이면 D(1101)로 나타낸다.

● 팩(Pack) 형식의 표현

- 10진수의 한 자리를 표현하기 위해서 존 영역 없이 4비트를 사용하는 형식
- 최하위 4비트에 부호를 표시
- 양수(+): 1100(C), 또는 1111(F)
- 음수(-): 1101(D)

팩 형식의 10진수 표현 형식



● 팩 형식으로 10진수를 표현하는 예

• +4213

0000	0100	0010	0001	0011	1100
0	4	2	1	3	C(+)

• -4213

0000	0100	0010	0001	0011	1101
0	4	2	1	3	D(-)

[참고] 10진수는 바이트에 표현하지만 1개의 10진수는 nibble(4비트)이므로 4자리의 10진수를 표현하려면 실제로는 부호까지 5개 nibble이 필요하지만 3바이트를 사용해야 하므로 첫 번째 nibble을 0으로 채운다.



더 알아보기

사람과 컴퓨터 시스템의 연산 특성

- 사람은 수치 표현을 위해 대부분 10진수를 사용한다.
- 컴퓨터는 bit를 사용하므로 2진수를 사용한다.
- 컴퓨터의 연산은 2진수의 연산을 기본으로 하지만 10진수를 위한 연산 처리도 가능하도록 지원한다.
- 10진수 연산을 위한 방법은 기계적 처리가 아닌 프로그램으로 처리한다.
- 따라서 2진수의 연산은 빠르지만 10진수 연산은 연산 처리를 위해 프로그램을 처리하여 결과를 얻어 내려면 처리 시간이 상대적으로 많이 걸린다.
- 10진수를 2진수로 표현하기 위해 4비트를 사용하게 되고, 4비트로 표현이 가능한 수의 범위가 0~15이므로 16진수의 형태로 나타난다.

2 2진수의 연산

2진수의 연산은 덧셈, 뺄셈, 곱셈, 나눗셈 등의 사칙 연산을 수행할 수 있지만 컴퓨터 시스템에서 사칙 연산은 덧셈 연산을 기본으로 처리한다. 컴퓨터 시스템에서 뺄셈은 보수를 구해 더해 주고, 곱셈은 덧셈을 반복하며, 나눗셈은 뺄셈을 반복함으로써 처리할 수 있다.

(1) 덧셈

2진수의 덧셈은 모든 자리의 합의 결과가 0이거나 1이 된다는 것을 제외하고는 10진수의 덧셈 원리를 이용하여 2를 단위로 자릿수가 하나씩 올라가는 규칙을 따른다.

2진수의 덧셈 규칙

0+0=1
0+1=1
1+0=1
1+1=10

→ 자리 올림

예제

2진수 1011과 0101을 더해 보자.

풀이

$$\begin{array}{r} 1011 \\ + 0101 \\ \hline 10000 \end{array}$$

$\therefore 1101_{(2)} + 0101_{(2)} = 10000_{(2)}$

(2) 뺄셈

컴퓨터 시스템에서 2진수의 뺄셈은 보수를 사용하여 덧셈으로 바꾸어 처리하며, 2진수에서 사용하는 보수의 종류에는 1의 보수와 2의 보수가 있다.

보수

보수는 기준이 되는 수에서 주어진 수의 값을 뺀 나머지 값을 말한다. 예를 들어, 주어진 수가 4인 경우 10진법에서 10의 보수는 10에서 4를 뺀 6이 되고, 9의 보수는 9에서 4를 뺀 5가 된다.

1 1의 보수와 2의 보수

1의 보수는 각 자리의 0을 1로, 1은 0으로 바꾸어 구하며, 2의 보수는 1의 보수에 1을 더하여 구한다.

예제

2진수 1010에 대한 1의 보수와 2의 보수를 구해 보자.

풀이

2진수 → 1010

1의 보수 → 0101

2의 보수 → 0110

1111 0은 1로, 1은 0으로 바꾼다.

↓ 구해진 1의 보수에 1을 더한다.

$$\begin{array}{r} 0101 \\ + 1 \\ \hline 0110 \end{array}$$

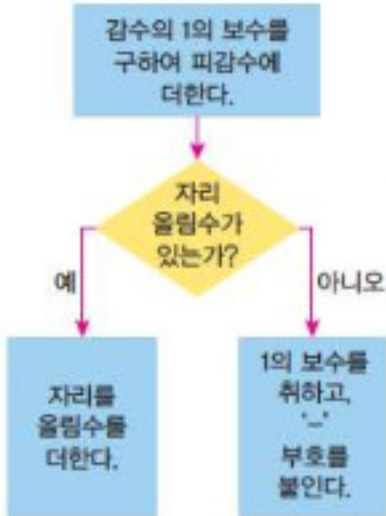


그림 1-8 1의 보수에 의한 뺄셈 과정

감수와 피감수

감수는 빼는 수를 말하고, 피감수는 뺄 수를 당하는 수이다.

2 1의 보수에 의한 뺄셈

1의 보수에 의한 뺄셈 과정은 그림 1-8과 같은 과정으로 한다. 즉, 감수의 1의 보수를 피감수에 더한다. 이때 자리 올림수가 발생하면 그 수를 구한 값에 더한다.

예제

2진수 뺄셈 $0111_{(2)} - 0100_{(2)}$ 과 $0100_{(2)} - 0111_{(2)}$ 을 1의 보수를 이용하여 구해 보자.

풀이

$$\begin{array}{r}
 0111 \\
 - 0100 \\
 \hline
 \end{array}
 \Rightarrow
 \begin{array}{r}
 0111 \\
 + 1011 \leftarrow 1\text{의 보수} \\
 \hline
 10010 \\
 + 1 \leftarrow \text{자리 올림수 1을 더한다.} \\
 \hline
 0011
 \end{array}$$

$$\begin{array}{r}
 0100 \\
 - 0111 \\
 \hline
 \end{array}
 \Rightarrow
 \begin{array}{r}
 0100 \\
 + 1000 \leftarrow 1\text{의 보수} \\
 \hline
 1100 \\
 - 0011 \leftarrow 1\text{의 보수} \\
 \hline
 0011
 \end{array}$$

3 2의 보수에 의한 뺄셈

2의 보수에 의한 뺄셈 과정은 그림 1-9와 같은 과정으로 한다. 즉, 감수의 2의 보수를 피감수에 더한다. 이때 자리 올림수가 발생하면 버리고, 없으면 연산 결과에 2의 보수를 취한 후 그 앞에 '-' 부호를 붙인다.

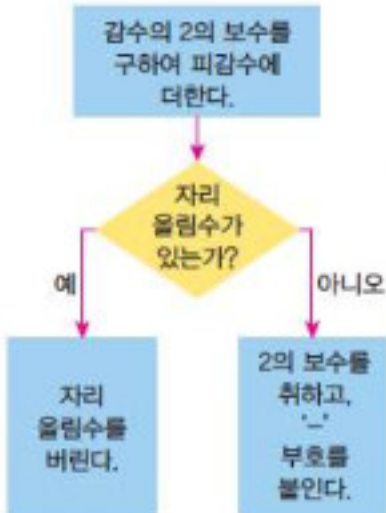


그림 1-9 2의 보수에 의한 뺄셈 과정

예제

2진수 뺄셈 $0111_{(2)} - 0100_{(2)}$ 과 $0100_{(2)} - 0111_{(2)}$ 을 2의 보수를 이용하여 구해 보자.

풀이

$$\begin{array}{r}
 0111 \\
 - 0100 \\
 \hline
 \end{array}
 \Rightarrow
 \begin{array}{r}
 0111 \\
 + 1100 \leftarrow 2\text{의 보수} \\
 \hline
 10011 \\
 \leftarrow \text{자리 올림수 1을 버린다.} \\
 \hline
 0111
 \end{array}$$

$$\begin{array}{r}
 0100 \\
 - 0111 \\
 \hline
 \end{array}
 \Rightarrow
 \begin{array}{r}
 0100 \\
 + 1001 \leftarrow 2\text{의 보수} \\
 \hline
 1101 \\
 - 0011 \leftarrow 2\text{의 보수} \\
 \hline
 0011
 \end{array}$$

(3) 곱셈

2진수의 곱셈 방법은 곱하는 수인 승수의 최하위 자리부터 승수가 1일 경우는 피승수를 더하고, 승수가 0일 경우는 피승수를 더하지 않는 과정을 되풀이한다. 자릿수가 커짐에 따라 곱해지는 수는 한 자리씩 자리올림을 한다. 이러한 2진수의 곱셈 규칙을 다음과 같이 나타낼 수 있다.

2진수의 곱셈 규칙

$$\begin{aligned} 0 \times 0 &= 0 \\ 0 \times 1 &= 0 \\ 1 \times 0 &= 0 \\ 1 \times 1 &= 1 \end{aligned}$$

2진수의 곱셈 규칙 예

$9 \times 5 = 45$	$1001 \times 101 = 101101$
$\begin{array}{r} 9 \\ \times 5 \\ \hline 45 \end{array}$	$\begin{array}{r} 1001 \text{ (피승수)} \\ \times 101 \text{ (승수)} \\ \hline 1001 \text{ (부분곱)} \\ 0000 \\ 1001 \\ \hline 101101 \text{ (부분곱의 합)} \end{array}$

(4) 나눗셈

2진수의 나눗셈 방법은 피제수에서 제수를 더 이상 뺄 수 없을 때까지 반복하는 방법을 사용하며, 뺄 수 있으면 몫은 1, 뺄 수 없으면 몫이 0이 된다. 이러한 2진수의 나눗셈 기본 법칙을 다음과 같이 나타낼 수 있다.

2진수의 나눗셈 규칙

$$\begin{aligned} 0 \div 0 &= 0 \\ 1 \div 1 &= 1 \end{aligned}$$

2진수의 나눗셈 규칙 예

$1100 \div 100 = 11$	$\begin{array}{r} 11 \\ 100 \overline{)1100} \\ \underline{100} \\ 100 \\ \underline{100} \\ 0 \end{array}$
----------------------	--

2진수의 나눗셈에서는 제수가 0인 경우에는 불능 상태가 되므로 제수가 1인 경우에만 나눗셈을 수행할 수 있다.



부동 소수점의 수의 산술은 가수와 지수의 연산을 분리해서 수행한다.

(1) 덧셈과 뺄셈

지수들이 동일한 값을 갖도록 가수의 소수점이 좌우로 이동한다. 다음으로 가수들 간의 덧셈 혹은 뺄셈을 수행 후 결과를 정규화한다.

(2) 곱셈

가수끼리는 곱셈 연산을 수행하고 지수끼리는 덧셈 연산을 수행한다. 그 후 마지막으로 결과 값을 정규화한다.

$$(0.1011 \times 2^3) \times (0.1001 \times 2^5)$$

$$\text{가수 곱하기: } 1011 \times 1001 = 01100011,$$

$$\text{지수 더하기: } 3 + 5 = 8,$$

$$\text{정규화: } 0.01100011 \times 2^8 = 0.1100011 \times 2^7$$

(3) 나눗셈

가수끼리는 나눗셈 연산을 수행하고 지수끼리는 뺄셈 연산을 수행한다. 그 후 마지막으로 결과 값을 정규화한다.

$$(0.1100 \times 2^5) \div (0.1100 \times 2^3)$$

$$\text{가수 나누기: } 1100 \div 1100 = 1,$$

$$\text{지수 뺄셈: } 5 - 3 = 2,$$

$$\text{정규화: } 0.1 \times 2^2$$

(4) 연산 과정에서 발생 가능한 문제

부동 소수점의 수의 표현 범위에서 오버플로우와 언더플로우가 발생하는 영역이 존재하는데, 연산의 결과가 이런 영역에 포함되면 오류가 발생한다.

- ① 지수 오버플로우 : 양의 지수 값이 최대 지수 값을 초과하여 발생하는 오류로 수가 너무 커서 표현될 수 없음을 의미하는 것으로 $+\infty$ 또는 $-\infty$ 로 나타낸다.
- ② 지수 언더플로우 : 음의 지수 값이 최대 지수 값을 초과하여 발생하는 오류로 수가 너무 작아서 표현될 수 없음을 의미하는 것으로 0으로 표시된다.
- ③ 가수 언더플로우 : 가수의 소수점 위치 조정 과정에서 비트들이 가수의 우측 편으로 넘치는 경우로서 반올림을 사용하여 문제를 해결한다.
- ④ 가수 오버플로우 : 같은 부호를 가진 두 가수들을 덧셈하였을 때 올림수가 발생하는 경우로 재조정 과정을 통하여 정규화하여서 해결한다.



활동 주제

고정 소수점 데이터 형식에서 2의 보수를 이용하여 4비트로 표현할 수 있는 수의 범위를 알아보자.

준비와 과정

- 모둠을 설정한다.
 - 4~5명의 모둠을 설정한다.
 - 모둠별로 모둠의 이름을 정하고, 모둠에서의 역할을 정한다.
- 모둠 이름: _____
- 모둠장: _____ 기록자: _____ 검색자: _____ 발표자: _____
- 부호 비트가 0인 경우를 알아본다.
- 부호 비트가 1인 경우를 알아본다.
- 4비트가 아닌 n비트로 확장하여 일반화시켜 본다.
- 정리한 자료를 발표자가 발표한다.
 - 다른 모둠이 발표할 때, 자신이 속한 모둠의 자료와 비교해 본다.

활동 내용

1. 부호 비트가 0인 양수의 최댓값을 찾아보자.

2. 부호 비트가 1인 음수의 최솟값을 찾아보자.

3. 일반화할 수 있도록 해보자.

디지털 정보의 표현

학습목표

- 컴퓨터 시스템에서 처리하는 문자 데이터의 표현 원리를 설명할 수 있다.
- 컴퓨터 시스템에서 처리하는 그림 데이터의 표현 원리를 설명할 수 있다.
- 컴퓨터 시스템에서 처리하는 소리 데이터의 표현 원리를 설명할 수 있다.
- 컴퓨터 시스템에서 처리하는 동영상 데이터의 표현 원리를 설명할 수 있다.
- 공개 소프트웨어를 이용하여 멀티미디어 데이터를 디지털 정보로 표현할 수 있다.

여는 이야기

세계 모든 언어를 사용할 수 있는 문자 코드

유니코드를 만들기 시작한 것은 90년대 초이지만 본격적으로 쓰기 시작한 것은 얼마 되지 않는다. 유니코드 표준은 아직도 진행형이다. 과거 각 나라마다 개발한 문자 코드로 자국의 언어와 영어를 사용하는 데 문제가 없었던 시절과 달리 인터넷이 보급되어 나라별 정보가 교환됨에 따라 자국어와 영어를 제외한 다른 나라 언어를 사용하는 경우에는 글을 쓸 수 없거나 출력되지 않는 문제가 발생하였다. 따라서 이러한 문제점을 해결하기 위해 지구상의 모든 언어를 하나의 포맷으로 통합하기 위한 표준 코드가 유니코드이다.

버전	연도	글자 수	주요 추가 사항
1.0.0	1991	7,161	기타카나, 구르무키 문자, 구지라트 문자, 그리스 문자, 데바나가리, 라오 문자, 라틴 문자, 밀라알람 문자, 벵골 문자, 아랍 문자, 아르메니아 문자, 오리아 문자, 조지아 문자, 주음부호, 칸나다 문자, 카탈 문자, 타밀 문자, 타이 문자, 텔루구 문자, 티베트 문자, 한글, 히라가나, 히브리 문자
2.0	1996	38,950	기존의 한글 완성자를 삭제하고, 11,172자의 새 한글 완성자 영역을 새로 지정함. 티베트 문자 새 위치에 추가, 서리게이트 영역을 지정하였고, 제15, 제16평면을 사용자 정의 영역으로 지정함.
3.0	1999	49,259	룬 문자, 몽골 문자, 베타 문자, 시리아 문자, 신할라 문자, 에티오피아 문자, 오걸, 이 문자, 점자, 체로키 문자, 캐나다 원주민 음절 문자, 크메르 문자, 타나 문자
4.0	2003	96,447	키프로스 문자, 캄부 문자, 선형 B 문자, 오스마냐 문자, 시위안 문자, 타이 렉 문자, 우가리트 문자
5.0	2006	99,089	발리 문자, 헤기 문자, 응코 문자, 파스파 문자, 페니키아 문자
6.0	2010	109,449	문자 3종류 추가, 플래잉카드 기호, 교통 및 지도 기호, 연금술 기호, 이모티콘과 예모지, 222자의 한중일 통합 한자 (CJK-D) 추가.
7.0	2014	113,021	문자 23종류 추가
8.0	2015	120,737	문자 6종류와 5771 한중일 통합 한자(CJK-E), 체로키 소문자
9.0	2016	128,237	서하 문자 등 6종의 문자

일상생활 속에서 사용하는 다양한 정보는 수치, 문자, 소리, 영상 등으로 이루어져 있다. 이러한 다양한 정보를 이루고 있는 데 이터들은 컴퓨터 시스템에서 어떻게 표현되는지 알아보고 이를 어떻게 가공하여 사용하는지 생각해 보자.

단/원/학/습/안/내

이 단원에서는 일상생활에서 사용하는 다양한 정보인 문자, 소리, 영상 등이 컴퓨터 시스템에서 어떻게 표현되는지 알아보고, 표현 방법에 따른 장단점을 찾아 실무에서 어떤 표현 방법이나 처리 방법을 사용하는 것이 바람직한지 익혀 컴퓨터 시스템 또는 디지털 정보 가공과 관련된 직무를 수행하는 데 필요한 실무 능력을 향상시킬 수 있도록 한다.

1 문자 데이터의 표현

사람들이 사용하는 한글, 영문자나 다른 외국 문자, 특수 문자 등을 컴퓨터 시스템에서 사용하려면 이들 문자를 구별하기 위해 미리 약속된 규칙에 따라 정해 놓은 문자 코드를 이용하여 컴퓨터 시스템이 처리할 수 있는 2진수 형태로 바꾸어 주어야 한다. 이때 사용하는 문자 코드로는 아스키 코드와 유니코드 등이 있다.

(1) 아스키코드

아스키코드(ASCII)는 미국 표준 협회가 컴퓨터 시스템 간에 정보 처리와 정보 교환을 위해 만든 코드로, 7비트로 구성되어 있다.

정보를 전송할 경우 기존의 아스키 코드에 오류를 검사할 수 있는 1비트(패리티 비트)를 추가해 총 8비트로 전송하면 전송 중에 발생하는 오류를 찾아낼 수 있다.

2진수	000	001	010	011	100	101	110	111	
0000	NUL	DLE	SP	0	@	P	1	p	
0001	SOH	DC1	!	1	A	Q	a	q	
0010	STX	DC2	"	2	B	R	b	r	
0011	ETX	DC3	#	3	C	S	c	s	
0100	EOT	DC4	\$	4	D	T	d	t	
0101	ENQ	NAM	%	5	E	U	e	u	
0110	ACK	SYN	^	6	F	V	f	v	
0111	BEL	ETB	&	7	G	W	g	w	
1000	BS	CAN	*	8	H	X	h	x	
1001	HT	EM	(	9	I	Y	i	y	
1010	LF	SUB	)	:	J	Z	j	z	
1011	VT	ESC	*	:	K	[	k	{	
1100	FF	FC	+	<	L		l	:	
1101	CR	QS	,	=	M	]	m	}	
1110	SO	RS	.	>	N		n	'	
1111	SI	US	/	?	O		o	DEL	

상위 비트

하위 비트

$100\ 0001_{(2)} = 65$
 상위 비트 하위 비트

그림 1-10 7비트 아스키 코드 표

문자 코드

컴퓨터 시스템에서 각 문자를 구별하기 위해 문자별로 약속된 2진수의 값

아스키코드와 유니코드의 비트 수와 문자 수 비교

이진 코드	비트 수	문자 수
아스키 코드	7비트	128개
유니 코드	16비트	65,536개

직무 따라잡기

그림 1-10에서 주어진 아스키코드 표를 보고 다음 문자열을 아스키코드로 표현해 보자.

$a=7.5$

(2) 유니코드



그림 1-11 유니코드의 예

유니코드(Unicode)는 전 세계의 문자를 컴퓨터 시스템에서 통일된 방법으로 표현할 수 있도록 설계된 문자 코드이다.

(3) 한글 표현

컴퓨터 시스템에서 한글 표현은 조합형 한글 코드와 완성형 한글 코드를 사용한다.

1 조합형 한글 코드

조합형 한글 코드는 초성, 중성, 종성으로 구분하여 코드를 부여하고, 초·중·종성을 조합하여 한글 문자를 표현하는 방식으로, 초성 5비트, 중성 5비트, 종성 5비트, 최상위 영/한 구분 1비트로 총 16비트를 부여한다.

2 완성형 한글 코드

완성형 한글 코드는 한글 글자에 1대 1로 코드를 부여하는 방식으로 한글 2,350자, 한자 4,888자, 특수 문자 986자에 코드를 부여한다.

예제

1. '컴퓨터'라는 문자열을 유니코드로 표현해 보자.

풀이

컴 : CEF4(1100 1110 1111 0100)

퓨 : D4E8(1101 0100 1110 1000)

터 : D130(1101 0001 0011 0000)

2. 한글 '물'을 완성형 한글 코드와 조합형 한글 코드로 표현해 보자.

풀이

완성형 물 ⇒ 1011101100111100

조합형

$\square + \tau + \text{ㄹ}$
 $\downarrow \quad \downarrow \quad \downarrow$
 01000 10100 01001

2

그림 데이터의 표현

그림 데이터는 그림을 구성하는 점인 화소의 집합으로 이루어진다. 그림을 흑백으로 나타낼 때에는 1점에 대해 명암을 나타내는 1비트가 필요하고, 그림을 컬러로 표현할 때에는 색과 명암을 표시하기 위해 많은 비트가 필요하다.

그림 데이터는 움직이는 영상이 아닌 정지 영상으로 그림, 사진, 도형 등을 말하며, 문자 데이터보다 쉽고 빠르게 정보를 표현할 수 있다. 컴퓨터 시스템에서 그림 데이터는 표현 방식에 따라 비트맵 방식과 벡터 방식으로 구분할 수 있다.

(1) 비트맵 방식

그림을 구성하는 최소 단위인 화소들이 모여 하나의 그림을 표현하는 방식으로 동일한 크기의 그림에서 화소 수를 늘리면 섬세하고 화려한 색상과 복잡한 이미지를 쉽게 표현할 수 있으나 화소 수가 많아짐에 따라 파일의 크기가 커지고 확대할 경우 이미지가 깨지는 현상이 나타난다. 화소는 컴퓨터 시스템에서 그림을 구성하는 최소 단위로 픽셀(pixel)이라고도 한다.



원본



128배 확대한 것

그림 1-12 비트맵 방식의 그림을 확대한 예



더 알아보기

해상도에 대해 알아보자.

해상도는 그림의 정밀도를 나타내는 단위로 1인치당 화소(ppi)나 점(dpi)의 수로 표시한다. 같은 크기의 그림에서 해상도가 높을수록 선명하고 깨끗하나 그림 파일의 용량이 커진다. 해상도는 이미지 해상도와 화면 해상도로 구분할 수 있다.

- (1) 이미지 해상도 : 하나의 그림(비트맵 방식)을 이루는 화소 수의 많고 적음을 나타낸다.
- (2) 화면 해상도 : 모니터 화면을 구성하는 화소 수를 나타낸다. 가로 화소 수와 세로 화소 수를 곱하는 형태로 표현한다.
(예, 1024×768, 1280×1024)

(2) 벡터 방식

벡터 방식

비트맵 방식에 비해 복잡한 그림을 만들기에는 어려움이 있다.

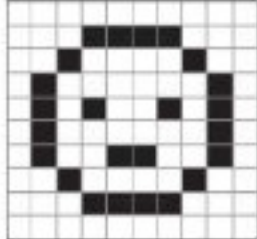
그림을 선과 좌표의 점들이 모여 선이나 도형의 형태로 표현하는 방식으로 비트맵 방식의 파일보다 파일의 크기는 작고, 그림을 확대하거나 축소해도 변형이 없다. 주로 글자(폰트), 로고, 그래프, 캐릭터를 디자인할 때 사용한다.



그림 1-13 벡터 방식의 이미지 표현

작업 따라잡기

다음은 비트맵 방식과 벡터 방식을 비교한 표이다. 빈 칸을 채워보자.

구분	비트맵 방식	벡터 방식
장점	실제와 가까운 그림 표현 가능 선명하고 정밀한 색상 표현	작은 파일 용량으로 정밀한 표현 가능
단점	 정밀도가 높을수록 파일 용량이 증가	정밀하고 다양한 그림의 표현이 어려움
표현 원리	 <pre> 0000000000 0001111000 0010000100 0100000010 0101001010 0100000010 0100110010 0100000010 0010000100 0001111000 </pre>	 <pre> ellipse 60, 20, 40, 40 rectangle 40, 50, 90, 100 fill line 10, 80, 60, 150 line 25, 140, 55, 130 </pre>



더 알아보기

그림 파일 형식

표현 방식	파일 형식	특징
비트맵 방식	GF	지원하는 색상의 수가 적은 만큼 파일 크기가 작아 인터넷 전송용으로 많이 사용한다.
	JPEG	정밀한 색의 표현이 가능하고 일반 이미지 파일 형식에 비해 파일 크기를 훨씬 작게 줄일 수 있어 인터넷에서 널리 사용한다.
	BMP	윈도 운영체제의 표준 파일 형식으로, 정밀한 표현이 가능하지만 파일의 크기가 크다는 단점이 있다.
	PNG	GF 포맷의 문제를 해결하고 개선하기 위해서 고안된 포맷으로 더 많은 색상을 지원하며 인터넷 상에 이미지 표현에 주로 사용된다.
벡터 방식	WMF	윈도 운영체제의 표준 벡터 방식 파일로, 클립아트와 같은 작은 크기의 그림 표현에 주로 쓰인다.
	AI	일러스트 작업을 하는 데 사용되는 파일 형식으로, 벡터 방식 표현에 가장 많이 쓰인다.

소리 데이터는 물체의 진동으로 생긴 파동이 공기를 통해 사람의 귀에 들리는 것으로 컴퓨터 정보 통신 기술이 발달함에 따라 사람의 말을 알아듣고, 또 사람에게 말을 하는 컴퓨터 시스템이 있다. 컴퓨터가 소리를 인식할 때에는, 소리의 장단, 고저, 강약, 음색에 따라 각기 다른 상태로 표현되기 때문에 어려운 점이 많다.

(1) 소리의 구성 요소

소리를 구성하는 요소로는 주파수, 진폭, 음색이 있다.

1 주파수

주파수는 1초에 반복되는 주기의 횟수를 말하며, 음의 높낮이를 나타낸다. 주파수가 높을수록 고음이 되며, 단위는 헤르츠(Hz)를 사용한다.

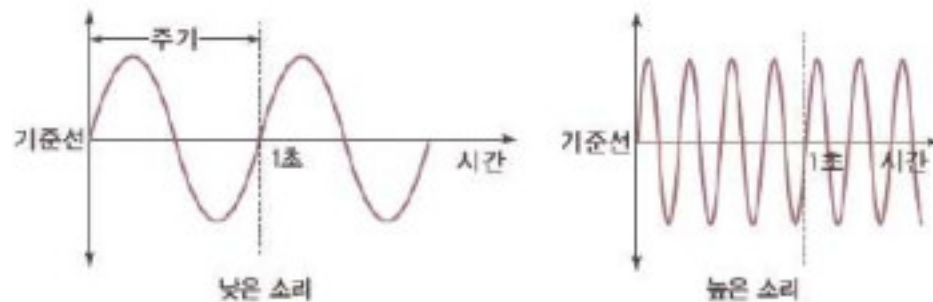


그림 1-14 주파수의 비교

표 1-2 아날로그 소리와 디지털 소리의 차이점

아날로그 소리	디지털 소리
• 편집이 어렵다. • 자연 그대로의 소리이다. • 음질이 비교적 깨끗하지 못하다. • 시간이 지나면 손상 가능성이 높다. • 악기 소리, 목소리가 해당된다.	• 컴퓨터로 편집 및 가공할 수 있다. • 자연에 없는 인위적 소리를 만들 수 있다. • 잡음이 없고 소리가 깨끗하다. • 시간이 지나도 손상되지 않는다. • CD나 MP3 소리가 해당된다.

2 진폭

진폭은 소리의 세기를 나타내는 요소로, 진폭이 클수록 소리가 커진다. 단위는 데시벨(dB)을 사용한다.

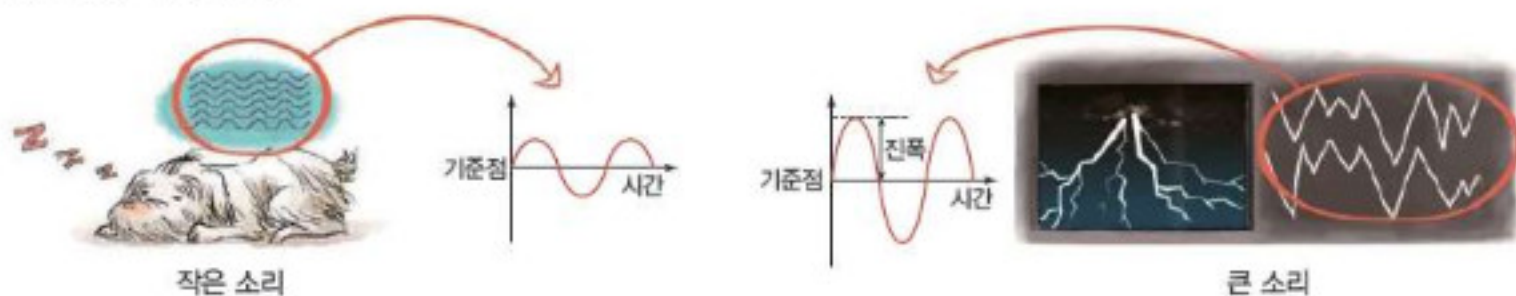


그림 1-16 진폭의 비교

주기

파형에서 같은 파형이 한 번 나타나는 데 걸리는 시간

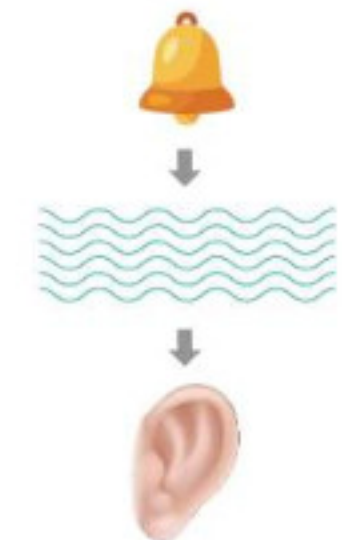


그림 1-15 소리의 전달 과정

3 음색

음색은 음파의 특성을 나타내는 것으로 같은 음을 내는 악기의 소리지만 서로 다른 느낌을 주는 것이 여기에 해당된다. 예를 들어 같은 주파수와 진폭을 가진 피아노와 바이올린의 소리가 전혀 다른 느낌을 주는 것은 피아노와 바이올린의 음색이 다르기 때문이다.



그림 1-17 음색의 비교

(2) 소리 데이터의 표현

양자화 펄스의 진폭

근접한 정수의 값으로 반올림한다.

컴퓨터 시스템에서 소리 데이터를 처리하기 위해서는 아날로그 형태인 소리 데이터를 디지털 형태로 변화하기 위해서는 표본화와 양자화, 부호화 과정을 거친다.



그림 1-18 소리 데이터의 표현 과정

1 표본화

표본화(샘플링)는 아날로그 신호를 일정한 시간 간격으로 잘라 내어 각 시간 간격마다 그 시간에 해당되는 진폭의 크기를 구한 것으로, 간격이 넓을수록 음질이 떨어지고, 간격이 좁을수록 높은 음질을 얻을 수 있다.

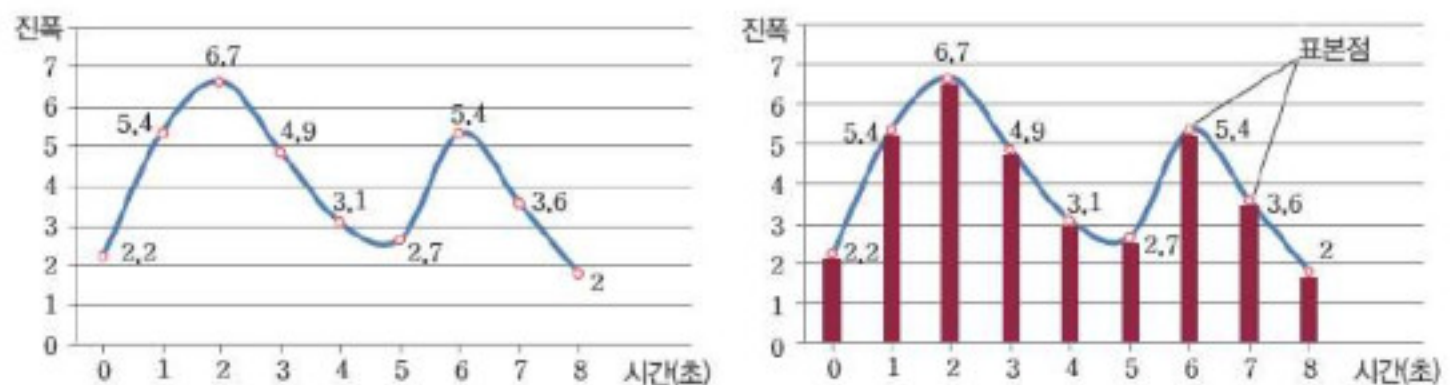


그림 1-19 표본화 (데이터 값은 바꿈)

2 양자화

양자화는 표본화에 의해 얻은 표본값을 디지털화할 수 있는 값으로 처리하는 과정이다. 신호의 크기를 몇 비트의 크기로 표현할 것인가를 결정하는 단계로 4비트로 양자화하면 신호의 진폭을 16단계로 표현되며, 16비트로 양자화하면 신호의 진폭을 65,536 단계로 표현된다.

3 부호화

부호화는 양자화된 신호들을 2진수로 표현하는 것으로, 신호의 크기가 6인 경우 4비트로 표현하면 '0110'이 된다.

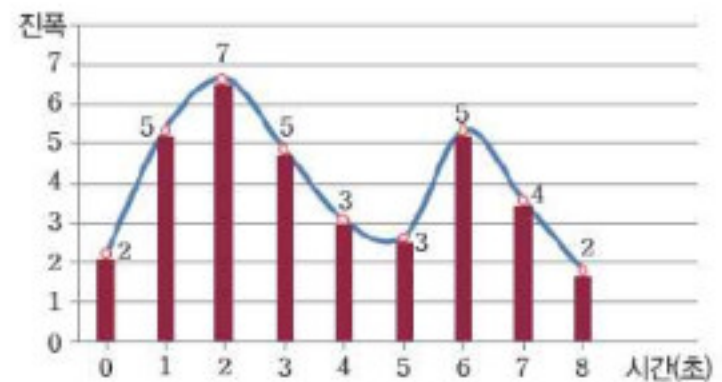


그림 1-20 양자화

예제

그림 1-20에서 양자화한 데이터를 3비트로 부호화하여 보자.

풀이

2	5	7	5	3	3	5	4	2
010	101	111	101	011	011	101	100	010



더 알아보기

최신 소리 데이터 파일의 형식

파일 형식	특징
WAV	개인용 컴퓨터의 대표적인 소리 저장 형식으로, 윈도 기반의 운영체제에서 사용하며, 소리를 압축하지 않고 저장한다.
MP3	소리를 압축하여 저장하는 형식으로, 파일 압축 효과가 좋고 우수한 음질을 얻을 수 있어 현재 다양한 매체에서 소리를 저장하거나 재생하는 데 많이 사용된다.
WMA	윈도 미디어 플레이어의 표준 형식으로, 인터넷 방송 및 웹을 통한 각종 동영상의 실시간 재생에 사용한다.
MD	소리를 전자 악기음의 정보로 기호화하여 저장한 형식으로, 다른 소리 형식에 비해 파일의 크기가 작다. 같은 미디 파일도 컴퓨터에 설치된 사운드 카드 종류에 따라 소리가 다르게 날 수 있다.
RMI	MD 파일 내용과 함께 곡에 대한 정보를 추가적으로 기록할 수 있도록 한 소리 파일 형식이다.
FLAC	오디오 데이터 압축을 위한 파일 형식으로 무손실 압축을 한다. 따라서 MP3 등과는 달리 오디오 스트림에 손실이 발생하지 않는다.

4 동영상 데이터의 표현

동영상은 변화하는 영상들이 담겨 있는 화면을 연속적으로 보여 주어 움직임을 표현하는 것으로, 화면의 움직임이 사람의 눈에 자연스러우려면 1초당 여러 화면을 움직여야 한다. 따라서 보기가 자연스러운 동영상 자료를 표현하기 위해서는 매우 빠른 중앙 처리 장치와 대용량의 저장 장치가 필요하다.

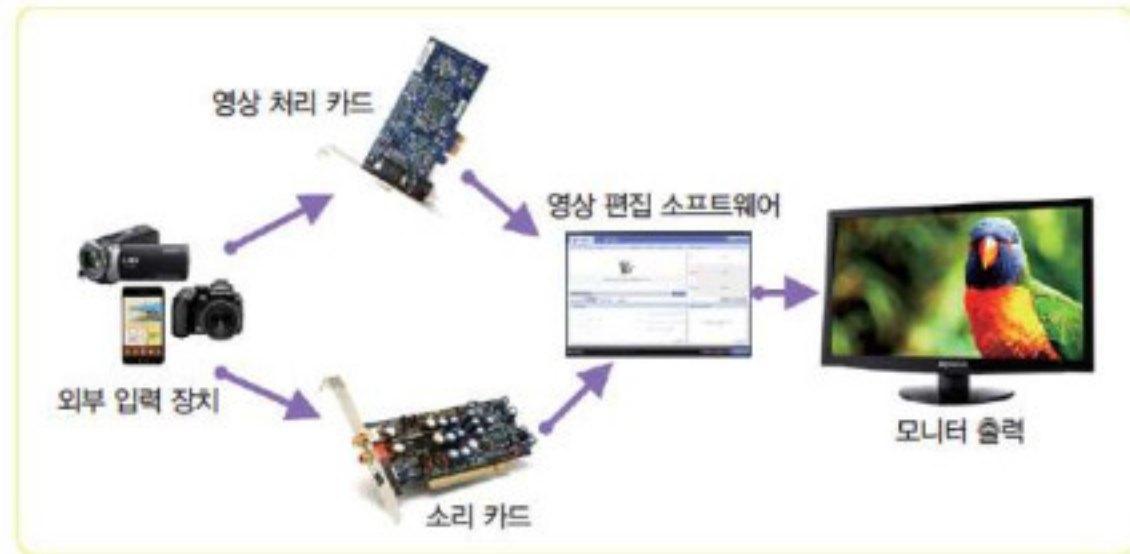


그림 1-21 컴퓨터 시스템에서 동영상의 처리 과정

(1) 동영상의 원리

잔상 효과

사람의 눈에 연속된 사진이나 그림을 빠르게 보여 주면 실제로 움직이는 것처럼 보이는 것

동영상의 원리는 사람이 갖는 잔상 효과를 이용한 것으로, 그림을 빨리 보여주어 움직이는 것처럼 보이게 하는 정지 영상의 모음이다. 이때 동영상을 구성하는 각각의 정지 화면을 프레임(frame)이라 하고, 1초당 보여주는 프레임의 개수를 프레임률(frame rate)이라 하며, 적정 프레임률은 15 ~ 30fps 정도이다.

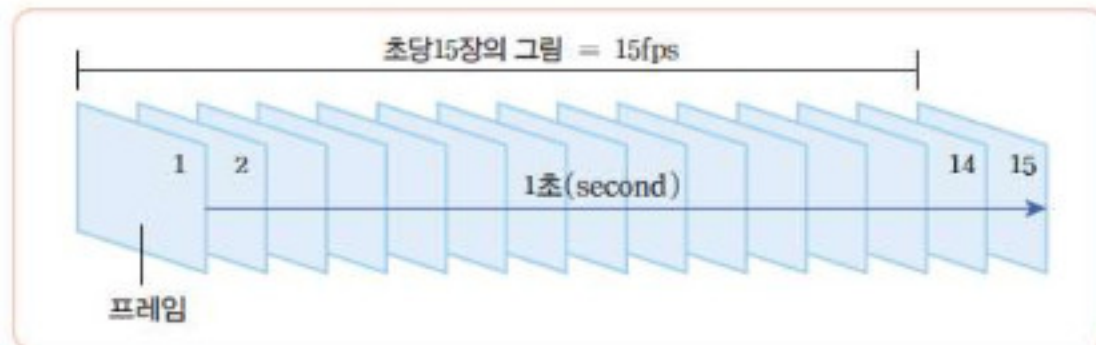


그림 1-22 동영상의 원리

(2) 동영상의 구성 요소

동영상은 비디오와 오디오로 구성되며, 동영상의 품질은 프레임 수, 해상도, 색상 수 등에 의해서 결정된다.

1 프레임 수

동영상 파일에서 프레임이 많을수록 더욱 자연스러운 화면을 볼 수 있으나, 프레임이 많아지면 파일의 용량이 커진다. 일반적으로 영상의 자연스러움을 유지하기 위해서는 15프레임 이상을 사용한다.

2 해상도

프레임을 구성하는 영상의 정밀도나 선명도를 나타내는 것으로, 가로 화소 수 × 세로 화소 수로 표현한다.

3 색상 수

동영상을 구성하는 프레임에 각각의 화소를 표현하기 위해 사용하는 색상의 수가 많을수록 화질은 좋아지지만, 파일의 크기가 커져 처리 시간이 오래 걸린다.

코덱(CODEC)

코더(coder)와 디코더(decoder)의 합성어로, 음성이나 비디오 데이터를 컴퓨터가 처리할 수 있게 디지털로 바꿔 주고, 그 데이터를 컴퓨터 사용자가 알 수 있게 모니터에 본래대로 재생시켜 주기도 하는 소프트웨어이다.

작업 따라잡기

1초 동안 재생되는 동영상 파일의 파일 크기 계산하는 방법을 알아보자.

[조건]

- 1초당 프레임 수 : 30 프레임
- 프레임의 크기 : 800×600
- 색상의 수 : 16 비트



더 알아보기

최신 동영상 데이터 파일의 형식

파일 형식	특징
AVI	윈도 기본 재생 동영상 형식으로, 재생할 때 압축을 해제할 수 있는 코덱이 필요하다.
MPEG	AVI와 함께 널리 쓰이는 형식으로, 압축률이 높아 CD와 방송용으로 많이 사용한다.
ASF	동영상을 네트워크에서 실시간으로 재생하는 형식이다.
MOV	애플사에서 개발한 동영상 파일 형식으로, 인터넷에서 실시간으로 재생이 가능하다.

5 멀티미디어 데이터의 가공

그림이나 소리, 동영상 자료 등 멀티미디어 자료는 바로 활용하는 경우도 있지만 대부분 원자료를 가공하여 사용하는 경우가 더 많다. 멀티미디어 자료를 가공할 경우에는 쉽게 가공할 수 있도록 도와주는 소프트웨어를 사용한다. 여기서는 운영체제에 포함된 프로그램이나 공개 소프트웨어를 사용하여 멀티미디어 자료를 가공하는 방법에 대하여 알아본다.

(1) 그림 데이터 편집

윈도 운영체제에 기본으로 포함된 '그림판'을 이용하여 그림 파일을 편집하여 보자.

1 그림 속성 알아보기

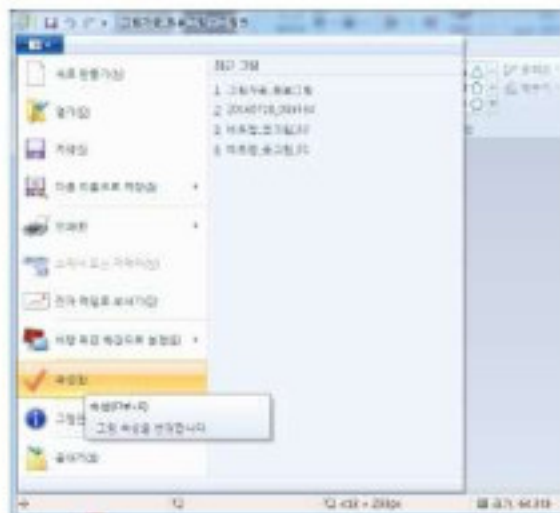
컴퓨터 시스템에서 사용하는 그림의 속성을 알아보자.

❶ [모든 프로그램]-[보조 프로그램]-[그림판]을 실행한다.

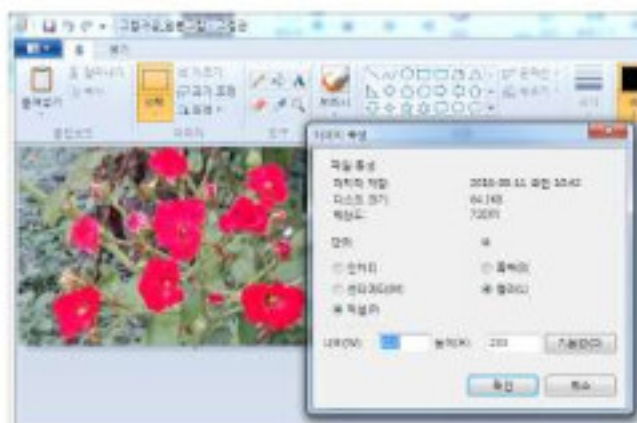


❷ 메뉴 [열기]를 실행한 후 나온 [열기] 대화상자에서 원하는 그림을 선택하여 불러온다.

❸ 메뉴 [속성]을 선택한다.



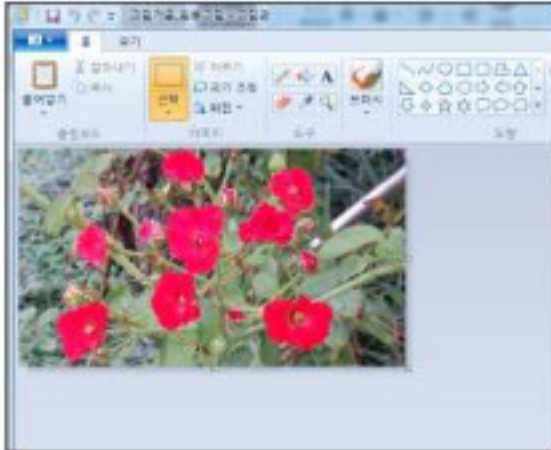
❹ [이미지 속성] 대화상자가 나오고 불러온 그림의 속성이 표시된다.



2 그림 크기 변경 및 파일 형식 변경하기

그림의 크기와 파일 형식을 변경하여 저장하여 보자.

❶ 그림판을 실행하고 변경하고자 하는 파일을 불러온다.

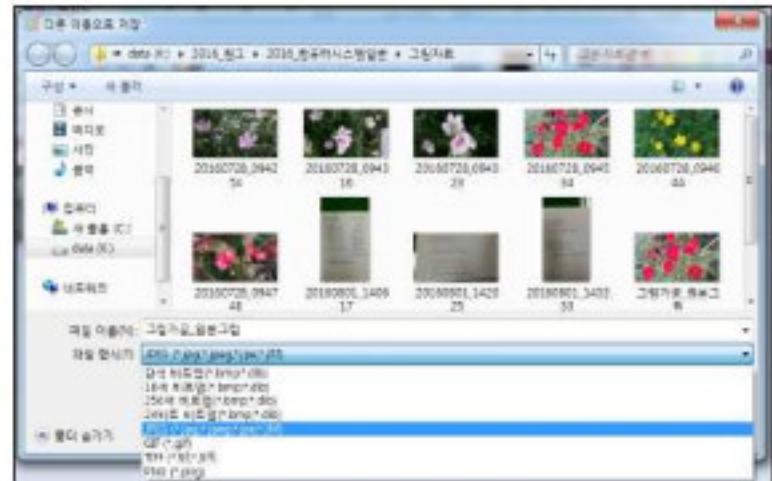


❷ 나온 [크기 조절 및 기울이기] 대화상자에서 '백분율' 또는 '픽셀'을 선택하여 그림의 크기를 조절한다.

❸ 도구상자에서 [크기 조절]을 클릭한다.



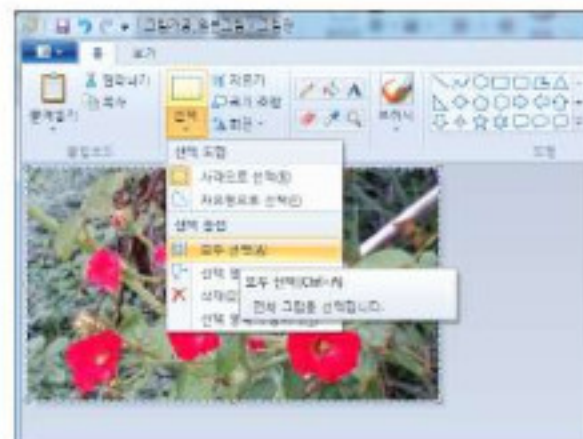
❹ 파일 형식의 변경하기 위해서는 메뉴 [다른 이름으로 저장]을 이용한다. 이 메뉴를 클릭하면 [다른 이름으로 저장] 대화상자가 나온다. 여기서 원하는 파일 형식으로 저장한다.



3 두 그림 합치기

❶ 그림판을 실행하고 합치고자 하는 첫 번째 그림 파일을 불러온다.

❷ 메뉴 [홈]에서 도구상자 [선택]을 클릭한 후 나온 하위 메뉴에서 [모두 선택]을 클릭한다.



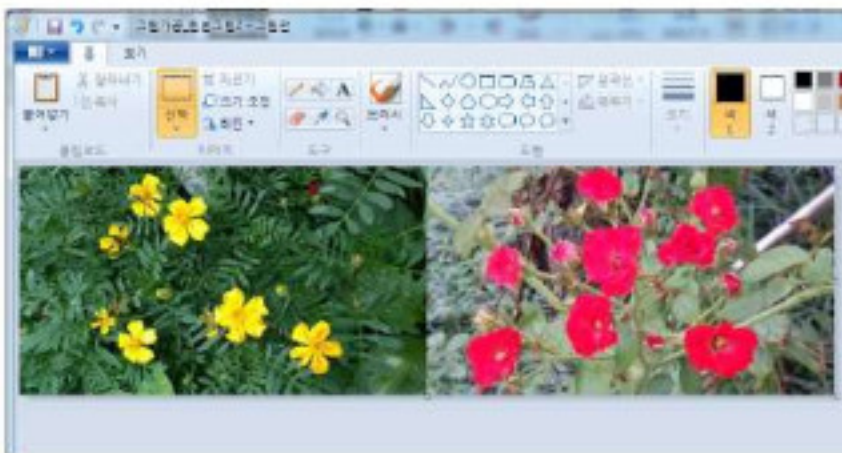
- ⑤ 메뉴 [복사]를 선택하거나 단축키 Ctrl+C을 사용하여 복사한 후, 두 번째 그림 파일을 불러온다.

- ⑥ 두 번째 그림이 올라온 후 앞에서 복사한 그림을 들어갈 수 있도록 크기 조절점을 변경하여 크기를 조절한다.



- ⑦ 메뉴 [홈]에서 도구상자 [붙여넣기]를 선택하거나 단축키 Ctrl+V키를 이용하여 붙여 넣으면 그림이 겹쳐진다. 붙여넣은 그림을 옆으로 이동하여 원하는 형태로 합친다.

- ⑧ 합친 그림 파일을 [다른 이름으로 저장] 메뉴를 이용하여 저장한다.



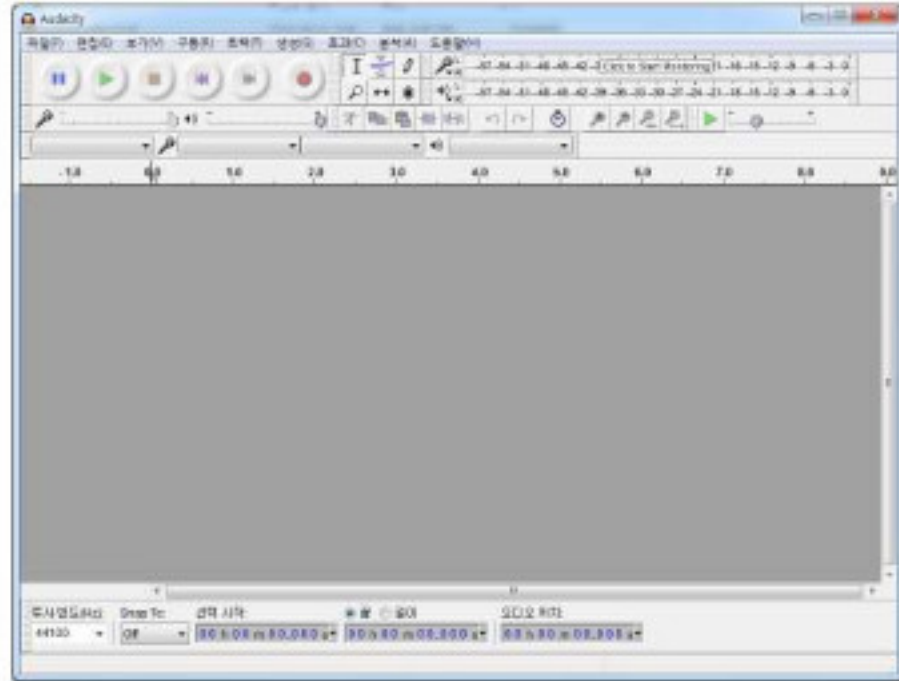
(2) 소리 데이터 가공

소리 데이터를 가공하기 위해서는 전문 장비와 기술이 필요하지만 간단히 편집하는 경우에는 무료로 사용할 수 있는 소리 편집 프로그램의 몇 가지 기능을 이용하면 된다. 여기서는 인터넷에서 프리웨어로 제공되는 'Audacity'을 이용하여 소리 데이터를 가공한다.

인터넷에서 프리웨어로 제공되는 'Audacity' 소리 편집 프로그램을 설치해 보자.

통이

- ❶ 인터넷에 접속하여 무료로 제공되는 오더시티(audacity) 프로그램을 다운로드 받는다.
- ❷ 다운로드 받은 파일을 실행하여 오더시티를 실행한다.
- ❸ 다음 그림과 같은 초기 화면이 실행된다.



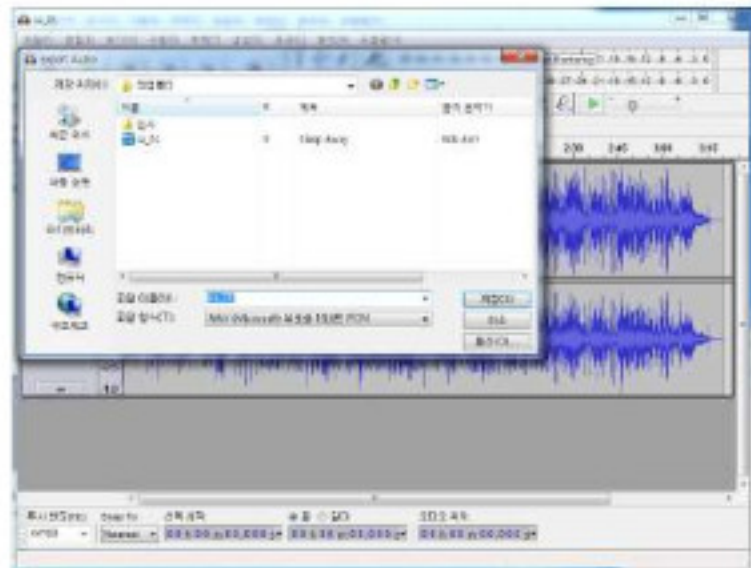
1 소리 데이터 만들기

소리 데이터는 필요에 따라 사람 목소리나 자연의 소리 등 아날로그 신호를 녹음하여 만들 수도 있고, 이미 만들어진 데이터를 가지고 새로운 데이터를 생성할 수도 있다.

내 목소리를 녹음하여 sa_01.wav 파일로 저장해 보자.

통이

- ❶ 소리 편집 프로그램을 실행한다.
- ❷ 녹음 버튼을 클릭한다.
- ❸ 마이크를 통해 내 목소리를 녹음한다.
- ❹ 메뉴 [파일]-[Export Audio...]를 선택한 후 나온 대화상자에서 파일 형식 'wav'를 선택하고 파일명 'sa_01'을 입력한 후 저장한다.



2 소리 데이터 편집하기

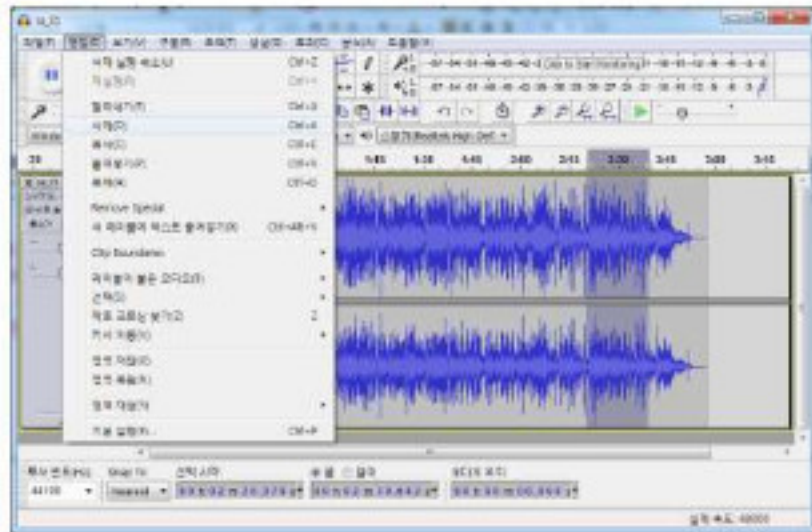
만들어진 소리 데이터는 필요 없는 부분을 잘라 내거나 특정 부분에 효과를 주거나 두 소리 파일을 합치는 등 다양하게 가공하여 사용할 수 있다.

실습하기

sa_01.wav 파일을 불러와 필요 없는 부분을 삭제한 후, sa_02.wav로 저장해 보자.

통이

- 1 소리 편집 프로그램을 실행한다.
- 2 메뉴 [파일]-[열기]를 클릭한 후 나온 대화상자에서 앞에서 만들어 놓은 소리 파일 'sa_01'을 불러온다.
- 3 소리를 재생한 후 편집창에서 잘라낼 부분을 마우스로 드래그 한다.
- 4 메뉴 [편집]-[삭제]를 선택하여 삭제한다. 메뉴 [편집]-[Remove Special]-[분할 잘라내기]를 하면 그 부분의 소리만 없어진다.
- 5 메뉴 [파일]-[Export Audio...]를 선택한 후 나온 대화상자에서 파일 형식 'wav'를 선택하고 파일명 'sa_02'로 저장한다.

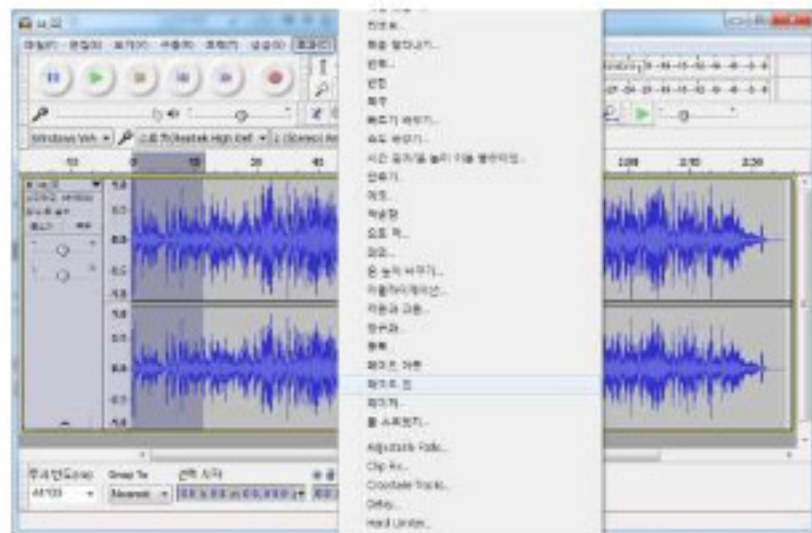


실습하기

sa_02.wav 파일을 불러와 시작 부분에 페이드 인, 끝부분에 페이드 아웃 효과를 넣어 sa_03.wav로 저장해 보자.

통이

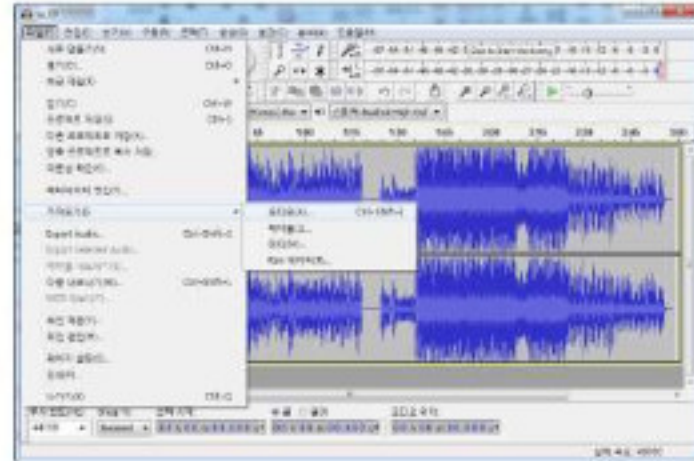
- 1 앞에서 만들어 놓은 소리 파일 'sa_02'를 불러온다.
- 2 페이드 인 효과를 줄 부분을 마우스로 드래그하여 선정한다.
- 3 메뉴 [효과]-[페이드 인]을 선택하여 페이드 인 효과를 준다.
- 4 페이드 아웃 효과를 줄 부분을 마우스로 드래그하여 선정한다.
- 5 메뉴 [효과]-[페이드 아웃]을 선택하여 페이드 아웃 효과를 준다.
- 6 파일명 'sa_03'으로 저장한다.



sa_03.wav파일을 불러와 배경 음악을 넣어 sa_04.wav로 저장해 보자.

동영상

- ❶ 앞에서 만들어 놓은 소리 파일 'sa_03'을 불러온다
- ❷ 배경 음악으로 사용할 파일을 불러오기 위해 메뉴 [파일]-[가져오기]-[오디오]를 선택한다.
- ❸ 배경 음악 파일을 불러오면 또 하나의 음악 트랙(실제 보이는 것은 R, L 트랙 두 개가 보인다.)
- ❹ 편집 모드에서 "시간 이동 도구" ↔ 커서로 변환하여 위 아래 소리 파일을 좌우로 움직여 필요한 위치에 놓는다.
- ❺ 파일명 'sa_04'로 저장한다.

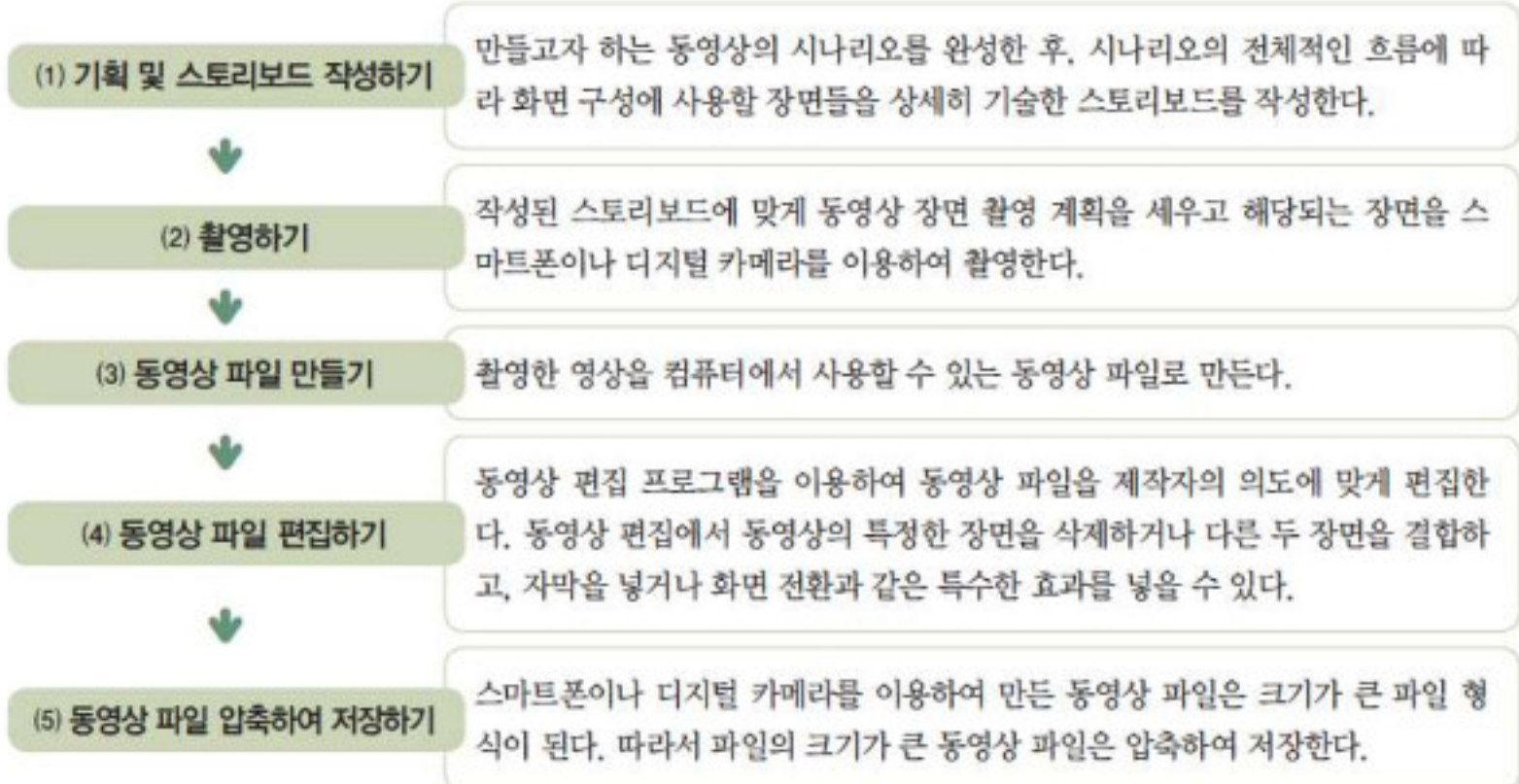


(3) 동영상 데이터 가공

동영상 데이터를 가공하기 위해서는 전문 장비와 기술이 필요하지만 간단히 편집하는 경우에는 동영상 편집 프로그램의 몇 가지 기능을 이용하면 된다. 동영상 편집 프로그램에는 여러 가지 종류가 있지만 여기서는 윈도 운영체제에서 무료로 사용할 수 있는 '무비메이커'를 이용하여 동영상 데이터를 가공한다.

동영상 제작 절차

동영상을 제작하는 과정을 살펴보면 다음 그림과 같다.



1 동영상 데이터 만들기

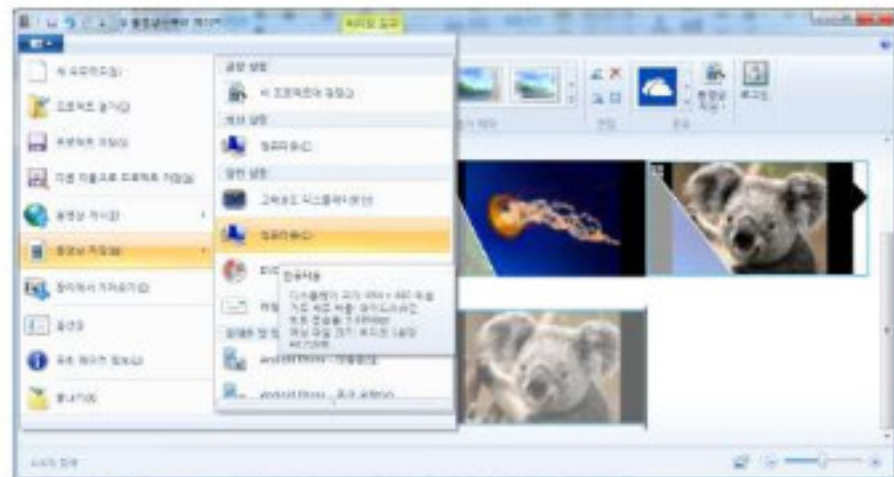
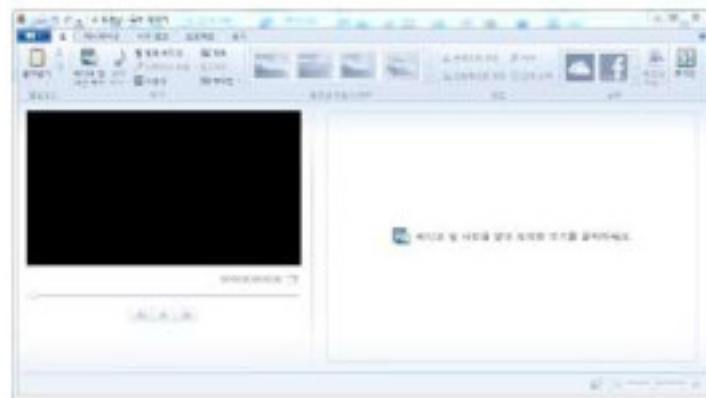
영화처럼 수준 높은 촬영 기법이나 특수 효과가 들어가 있는 영상을 만들기는 힘들지만 스마트폰이나 디지털 카메라, 컴퓨터를 이용하여 간단한 동영상 데이터를 제작할 수 있다.

실습하기

여러 장의 사진을 가져와 동영상 파일로 만들어 보자.

준비

- ① 작업에 들어가기 전에 필요한 것을 준비한다.
 - 작업에 필요한 사진과 동영상을 한 폴더에 넣어둔다(작업 시간과 효율을 높일 수 있다).
 - 스토리텔링을 통한 스토리보드를 작성한다(제작 동영상의 주제에 알맞은 사진과 배경 음악 등을 미리 선정한다).
- ② 동영상 편집 프로그램 무비메이커를 실행한다.
- ③ 메뉴[홈]에서 [비디오 및 사진 추가] 아이콘을 이용하여 동영상 파일로 만들고자하는 사진들을 불러와 원하는 순서로 배열한다.
- ④ 메뉴 [애니메이션]을 이용하여 불러온 사진에 화면 전환과 이동 및 확대/축소 애니메이션 효과를 넣는다.
- ⑤ 메뉴 [동영상 저장]-[컴퓨터용]을 이용하여 동영상 파일을 저장한다. 만일 작업이 끝나지 않아서 다음에 다시 작업을 할 경우에는 [프로젝트 저장] 메뉴를 사용한다.



2 동영상 데이터 편집하기

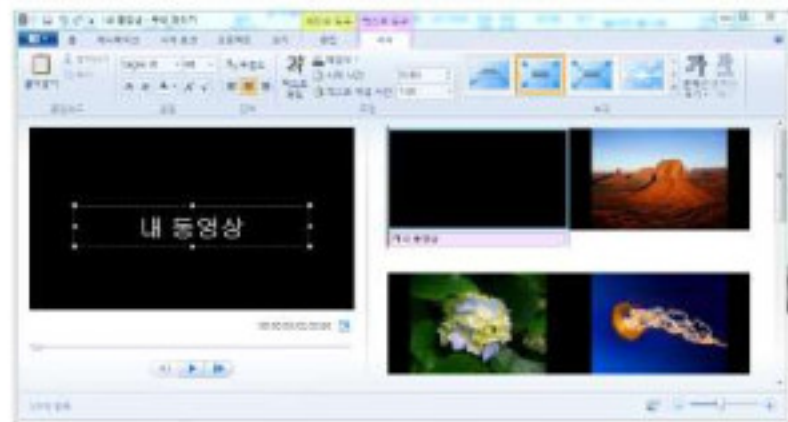
만들어진 동영상 데이터에 제목이나 자막, 배경 음악 등을 넣어 다양하게 가공하여 사용할 수 있다.

실습하기

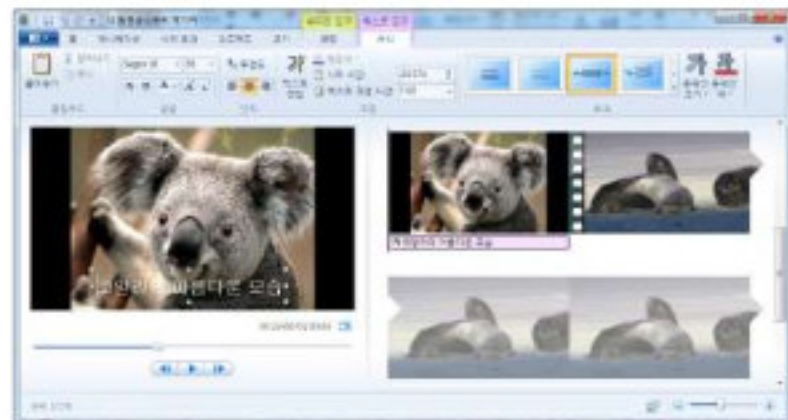
제작한 동영상에 제목과 설명글을 넣어 저장하여 보자.

동영상

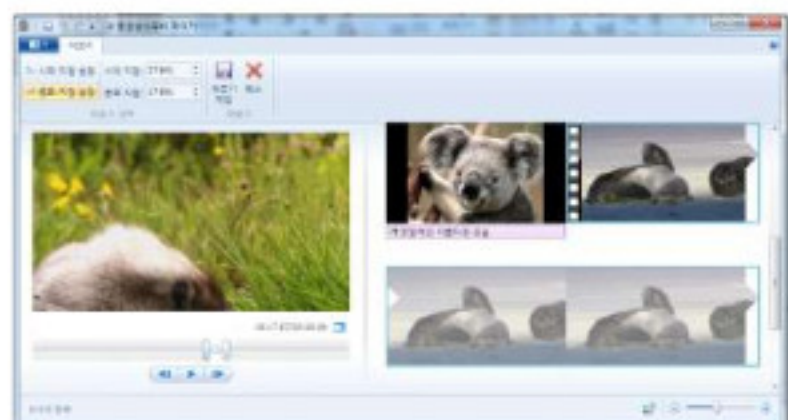
- ❶ 무비메이커 프로그램을 실행하고, 제목을 넣을 동영상을 불러온다.
- ❷ 메뉴 [홈]을 클릭한 후 [제목] 아이콘을 클릭하면 오른쪽과 같이 제목표시줄에 [텍스트 도구]가 나오고, 메뉴표시줄에 [서식] 메뉴가 나온다.
- ❸ 제목을 입력한 후, [효과]를 선택하여 제목에 대한 효과를 넣어준다.



- ❹ 설명글을 넣을 프레임을 선택한 후, 메뉴 [홈]을 클릭한 후 [자막] 아이콘을 클릭하여 나온 창에서 설명글을 입력한다.



- ❺ 만일 필요 없는 부분을 잘라내는 방법은 다음과 같다.
 - 메뉴 [편집]을 클릭한 후 [자르기 도구] 아이콘을 클릭하면 오른쪽과 같은 [자르기] 창이 나온다.
 - [자르기] 창에서 왼쪽 화면의 슬라이더 바를 이동하여 자를 시작 지점과 종료 지점을 설정한 후 [자르기 저장] 아이콘을 클릭한다.
- ❻ 같은 방법으로 필요한 부분에 설명할 글을 입력한 후 메뉴 [동영상 저장]-[컴퓨터용]을 이용하여 동영상 파일을 저장한다.

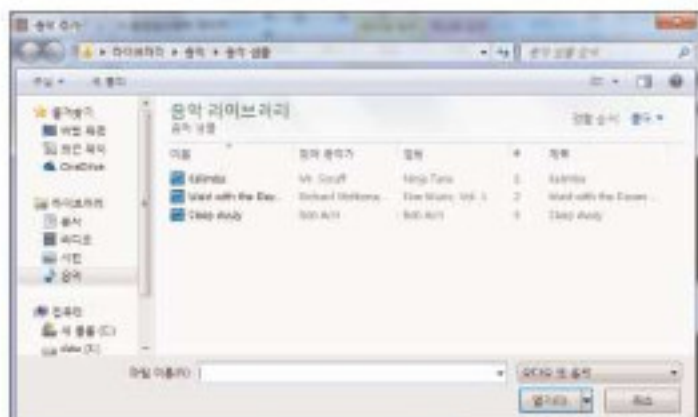


실습하기

제작한 동영상에 배경 음악을 저장하여 보자.

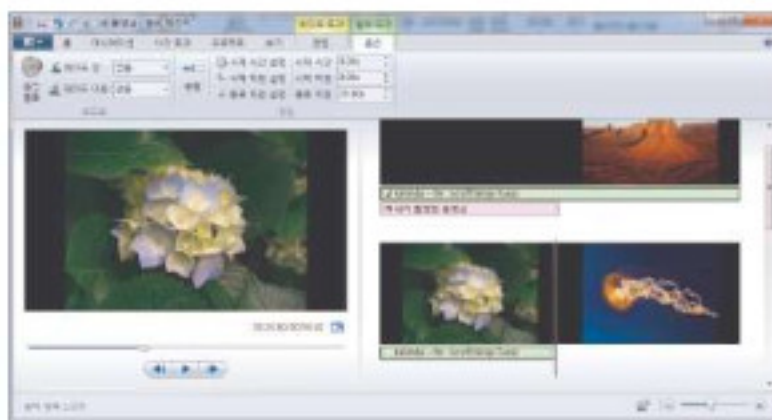
풀어

- 1 무비메이커 프로그램을 실행하고, 배경 음악을 넣을 동영상을 불러온다.
- 2 메뉴 [홈]을 선택한 후 [음악 추가]-[PC에서 음악 추가]-[음악 추가] 아이콘을 클릭하면 다음과 같은 [음악 추가] 대화상자가 나온다. 여기서 배경 음악으로 사용할 음악을 선택한다.



참고 : 저작권이 있는 음악을 사용할 때에는 개인적으로 소장만 가능하다. 저작권 걱정 없이 사용할 수 있는 MP3 음악을 이용할 경우 무비메이커에서는 [Free Music Archive]를 선택하여 사용한다.

- 3 배경 음악을 선택하면 무비메이커 창에 오른쪽 그림과 같이 제목표시줄에 [음악 도구]가 나오고, 메뉴표시줄에 [옵션] 메뉴가 나온다.
- 4 여기서 배경 음악을 넣을 위치인 [시작 지점 설정] 및 [종료 지점 설정]을 한다. 또한 페이드 인이나 페이드 아웃 효과를 넣어 주는 것도 여기서 선택하면 된다.
- 5 같은 방법으로 필요한 부분에 배경 음악을 넣은 후 메뉴 [동영상 저장]-[컴퓨터용]을 이용하여 동영상 파일을 저장한다.



읽을 거리

데이터의 압축

압축은 데이터를 더 작은 저장 공간에 효율적으로 기록하기 위한 기술로, 크기가 큰 데이터를 더 작은 크기로 변환시키는 인코딩 과정과 저장된 데이터를 다시 불러와 원래 데이터 형태로 복원시키는 디코딩 과정으로 이루어진다. 데이터가 압축된 정도를 압축률이라 하며, 압축률은 압축한 데이터의 크기를 인코딩하기 전의 데이터 크기로 나눈 것이다. 그러므로 압축률은 0과 1 사이의 값을 가지며, 값이 0에 가까울수록 더 많은 압축이 일어났음을 의미한다.

$$\text{압축률} = \frac{\text{압축한 데이터의 크기}}{\text{압축 전 데이터의 크기}}$$

데이터 압축 방법에는 압축 과정에서 원래의 정보를 잃지 않고 압축하는 무손실 압축 방법과, 원래의 정보를 일부 잃으면서 압축하는 손실 압축 방법이 있다.



활동 주제

산업의 한 분야를 소개하는 동영상을 만들어 보자.

준비와 과정

- 모둠을 설정한다.
 - 4~5명의 모둠을 설정한다.
 - 모둠별로 모둠의 이름을 정하고, 모둠에서의 역할을 정한다.
- 모둠 이름: _____
- 모둠장: _____ 기록자: _____ 검색자: _____ 발표자: _____
- 기획 및 스토리보드 작성하기
 - 동영상 제작에 필요한 사진 및 기초 동영상 촬영하기
 - 배경 음악 삽입 및 동영상 편집하기
 - 정리한 자료를 발표하기
 - 다른 모둠이 발표할 때, 자신이 속한 모둠의 자료와 비교해 본다.

활동 내용

1. 기획 및 스토리보드를 작성한다.

2. 동영상 제작에 필요한 사진 및 기초 동영상을 촬영한다.

2. 배경 음악 삽입 및 동영상을 편집한다.

01. 수의 체계

- (1) 자료 : 현실 세계에서 관찰이나 측정을 통하여 수집한 사실
- (2) 정보 : 어떤 일을 하는 데 필요한 자료들만 모아 놓거나 정리한 것으로, 컴퓨터를 통하여 자료들을 새로운 내용으로 정리하거나 가공하여, 보다 새로운 가치를 지니도록 한 자료의 집합
- (3) 진법 : 수를 표시하는 방법 중의 하나로, 숫자의 위치에 따라 가중치가 달라지는 수로 나타내는 방법

10진법	0부터 9까지 10개의 숫자를 사용하여 수를 표현하는 방법. 10진수는 10진법에 의해 표현된 수
2진법	0과 1의 2개의 숫자를 사용하여 수를 표현하는 방법. 2진수는 2진법에 의해 표현된 수
8진법	0부터 7까지 8개의 숫자를 사용하여 수를 표현하는 방법. 8진수는 8진법에 의해 표현된 수
16진법	0부터 15까지 16개의 숫자를 사용하여 수를 표현하는 방법으로, 10부터 15까지는 10은 A, 11은 B, 12는 C, 13은 D, 14는 E, 15는 F로 각각 표현. 16진수는 16진법에 의해 표현된 수

02. 디지털 정보의 연산

(1) 진법 변환

- ① 10진수에서 2진수, 8진수, 16진수로 변환
 - 정수 부분의 변환 방법 : 10진수의 정수를 변환할 때는 변환하는 진수의 밑수로 나누어 몫과 나머지를 구하여 그 몫이 0이 될 때까지 계속 나눈 후, 구해진 나머지를 역순으로 표현
 - 소수 부분의 변환 방법 : 10진수의 소수 부분을 변환할 때는 변환하고자 하는 진수의 밑수로 소수점 이하가 0이 될 때까지 곱한 후, 정수 부분을 취하여 자리올림수가 발생한 순서대로 표현
- ② 2진수에서 8진수로의 변환 : 2진수의 세 자리는 8진수의 한 자리로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 세 자리씩 묶어서 각 묶음에 대응되는 8진수의 한 자리로 표현
- ③ 2진수에서 16진수로의 변환 : 2진수의 네 자리는 16진수의 한 자리로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 네 자리씩 묶어서 각 묶음에 대응되는 16진수의 한 자리로 표현
- ④ 8진수에서 2진수로의 변환 : 8진수의 한 자리는 세 자리의 2진수로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 8진수의 한 자리를 세 자리의 2진수로 표현
- ⑤ 16진수에서 2진수로의 변환 : 16진수의 한 자리는 네 자리의 2진수로 표현하므로, 소수점을 중심으로 왼쪽과 오른쪽으로 각각 16진수의 한 자리를 네 자리의 2진수로 표현

(5) 2진수의 덧셈

2진수의 덧셈은 10진수의 덧셈과 같이 각 자리의 수를 더하고, 두 개의 수가 모두 1일 때 자리올림

(6) 2진수의 뺄셈

① 1의 보수에 의한 뺄셈

[1단계] 감수에 대한 1의 보수를 구한다.
 [2단계] 피감수와 감수의 보수 값을 더한다.
 [3단계] 연산 결과 자리올림수가 발생하면 최하위 자리에 1을 더하고, 자리올림수가 없으면 연산 결과에 1의 보수를 취한 후 그 앞에 '-' 부호를 붙인다.

② 2의 보수에 의한 뺄셈

[1단계] 감수에 대한 2의 보수를 구한다.
 [2단계] 피감수와 감수의 보수 값을 더한다.
 [3단계] 연산 결과 자리올림수가 발생하면 올림수를 버리고, 자리올림수가 없으면 연산 결과에 2의 보수를 취한 후 그 앞에 '-' 부호를 붙인다.

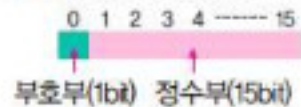
03. 디지털 정보의 표현



(1) 수치 데이터 표현

① 고정 소수점 데이터 형식

• 2바이트(16비트)정수형



• 4바이트(32비트)정수형



부호부
양수(+)이면 '0', 음수(-)이면 '1'을 표현한다.

정수부
기억할 숫자를 2진수로 변환하여 표현한다.

② 부동 소수점 데이터 형식

• 4바이트(32비트)단정도 실수형



부호부
양수(+)이면 '0', 음수(-)이면 '1'을 표현한다.

지수부
지수를 2진수로 지정한다.

가수부
소수점 아래 10진 유효숫자를 2진수로 변환하여 저장한다.

(2) 문자 데이터

- ① 아스키코드 : 미국 표준 협회가 컴퓨터 간에 정보 교환을 위해 만든 표준 코드, 7비트로 구성되어 있어, 총 $128(2^7)$ 개의 문자를 표현
- ② 유니코드 : 세계 모든 나라의 언어를 통일된 방법으로 표현할 수 있도록 제안된 국제 코드로 세계 여러 언어의 문자와 그 밖의 기호들도 표현 가능
- ③ 초성, 중성, 종성으로 구분하여 코드를 부여하는 조합형 한글 코드와 한글 글자에 1대 1로 코드를 부여하는 완성형 한글 코드를 사용하여 한글 표현
- (3) 소리 데이터 : 컴퓨터에서 아날로그 형태인 소리 데이터를 표본화, 양자화, 부호화 과정을 거쳐 디지털 형태로 변환하는 과정
- (4) 영상 데이터 : 영상 데이터는 그림을 구성하는 화소(픽셀)의 집합으로 이루어져 있으며, 동영상은 변화하는 영상들이 담겨 있는 화면을 연속적으로 보여 주어 움직임을 표현한 것



01 정보와 관련된 다음 설명으로 옳은 것을 <보기>에서 모두 고른 것은?

보기

- ㉠ 정보는 활용할 경우에 그 가치를 발휘한다.
- ㉡ 정보는 필요한 시점에 제공될 때 그 가치가 있다.
- ㉢ 정보는 사용자에게 관계없이 그 가치는 항상 일정하다.
- ㉣ 정보는 자료를 수요자가 원하는 형태로 가공한 것이다.

- ① ㉠, ㉡
- ② ㉢, ㉣
- ③ ㉠, ㉡, ㉢
- ④ ㉠, ㉡, ㉣
- ⑤ ㉡, ㉢, ㉣

02 10진수 12는 16진수로 얼마인가?

- ① B
- ② C
- ③ D
- ④ F
- ⑤ 10

03 2진수 $01101101_{(2)}$ 의 1의 보수는?

- ① $10010010_{(2)}$
- ② $10010110_{(2)}$
- ③ $11110010_{(2)}$
- ④ $10011111_{(2)}$
- ⑤ $01101101_{(2)}$

04 다음은 (가) 진수를 (나) 진수로 변환하는 과정을 나타낸 것이다. (가), (나)에 들어갈 진수를 바르게 짝지은 것은?

보기

- 소수점을 중심으로 왼쪽과 오른쪽으로 4자리씩 묶는다.
- 모자라는 자리는 0으로 채운다.
- 각 묶음마다 각 자리에 해당되는 가중치를 곱하여 이들을 더하여 표현한다.

- | (가) | (나) | (가) | (나) |
|-----|-----|-----|-----|
| ① 2 | 8 | ② 2 | 10 |
| ③ 2 | 16 | ④ 8 | 10 |
| ⑤ 8 | 16 | | |

05 다음 설명에 해당하는 것은?

이것이 높을수록 그림을 구성하는 화소의 수가 많아져 더욱 선명하고 깨끗하게 그림을 표현한다.

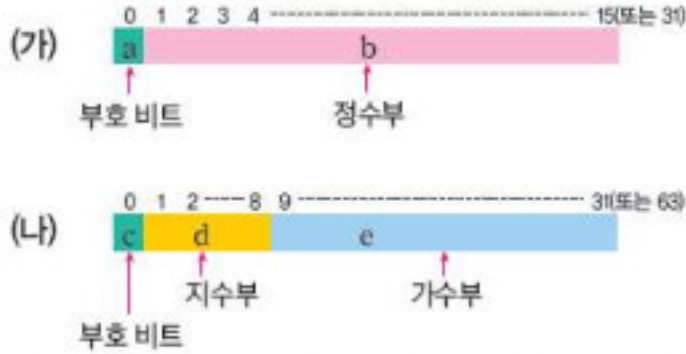
- ① 음색
- ② 픽셀
- ③ 비트맵
- ④ 해상도
- ⑤ 주파수

06 영희는 부동 소수점 데이터 형식에 대한 수행 평가의 단답형 문제를 다음과 같이 풀었다. 정답인 경우에는 2점 득점, 오답인 경우에는 1점 감점일 때 영희가 얻은 점수는?

- 아주 작은 수와 아주 큰 수의 표현에는 부적합하다. (○)
- 과학이나 공학 또는 수학적인 응용에 주로 사용되는 수 표현이다. (○)
- 일반적으로 고정 소수점 표현 방법보다 많은 기억 장소를 차지한다. (○)
- 고정 소수점 표현 방법보다 표현의 정밀도를 높일 수 있다. (○)

- ① 0점
- ② 1점
- ③ 2점
- ④ 4점
- ⑤ 5점

07 다음은 수치 데이터의 표현 방법의 일부를 나타낸 것이다.
이에 대한 설명으로 옳지 않은 것은?



- ① a, c에 들어가는 값은 음수이면 1이고, 양수이면 0이다.
- ② b에 들어가는 값은 10진수가 2진수로 변환되어 들어간다.
- ③ (가)는 정수를 (나)는 소수점이 있는 실수를 표현할 때 사용한다.
- ④ d의 8비트에는 지수의 음수를 나타내는 또 다른 부호 비트가 있다.
- ⑤ e는 10진 유효 숫자를 2진수로 변환하여 주어진 비트 수만큼 표현한다.

08 아날로그 소리 데이터를 일정한 간격으로 잘라 디지털 데이터로 변환하는 과정은?

- ① 표본화 ② 양자화
- ③ 부호화 ④ 이진화
- ⑤ 선형화

09 다음의 소수점 값이 있는 10진수를 다른 진수로 변환해 보자.

- ① 14.5를 2진수로 변환
- ② 14.25를 8진수로 변환
- ③ 22.125를 16진수로 변환

10 다음을 계산해 보자.

- ① $1011_{(2)} \times 0110_{(2)}$
- ② $1011_{(2)} \div 11_{(2)}$

11 사진이나 그림을 크게 확대하면 많은 점 또는 사각형을 볼 수 있는데, 이것을 ()이라고 한다. 이것은 그림을 구성하는 최소 단위이다.

12 2진수의 뺄셈 $11010_{(2)} - 11001_{(2)}$ 을 1의 보수와 2의 보수를 사용하여 계산한 후, 결과 값을 비교해 보자.

13 다음과 같은 형식의 그림 파일과 소리 파일들이 주로 사용되는 곳과 파일 형식의 특징을 조사해 보자.

구분	파일 형식	사용하는 곳	특징
그림	gif		
	jpg		
	wmf		
소리	wav		
	mp3		
	wma		