Chapter 4. 증명법

개요

- 증명의 일반적인 방법론을 고찰하고 여러 가지 증명법들을 살펴봄
- 공학이나 컴퓨터 관련 학문에 있어서 주어진 문제를 해결하기 위한 단계적 접근 방법을 제시함
- 공학과 관련된 엄밀한 입증과 증명 방법론을 고찰함
- 증명법의 종류
  - 수학적 귀납법, 모순 증명법, 직접 증명법, 대우 증명법, 존재 증명법, 반례 증명법, 필요충분조건 증명법 등

CONTENTS

4. 증명법

- 추론을 통한 수학적 증명은 대부분의 사람들에게 어렵게 느껴질 수 있음
- 그러나 증명 과정을 통하여 공학이나 수학을 비롯한 여러 분야에서 논리적 바탕에 기반을 둔 학문적 탐구가 가능함

4.1 증명의 방법론

정의 4-1 증명(proof)이란 논리적 법칙을 이용하여 주어진 가정으로부터 결론을 유도해내는 추론의 한 방법으로서, 어떠한 명제나 논증이 적절하고 타당한지를 입증하는 작업이다.

- 공학이나 컴퓨터 관련 학문에 있어서 주어진 문제를 해결하기 위해서는 증명의 단계적 접근 방식이 매우 효과적임

4.1 증명의 방법론

정의 ❹-1 증명(proof)이란 논리적 법칙을 이용하여 주어진 가정으로부터 결론을 유도해내는 추론의 한 방법으로서, 어떠한 명제나 논증이 적절하고 타당한지를 입증하는 작업이다.

증명의 단계적 접근 방법

1. 아이디어 스케치 단계

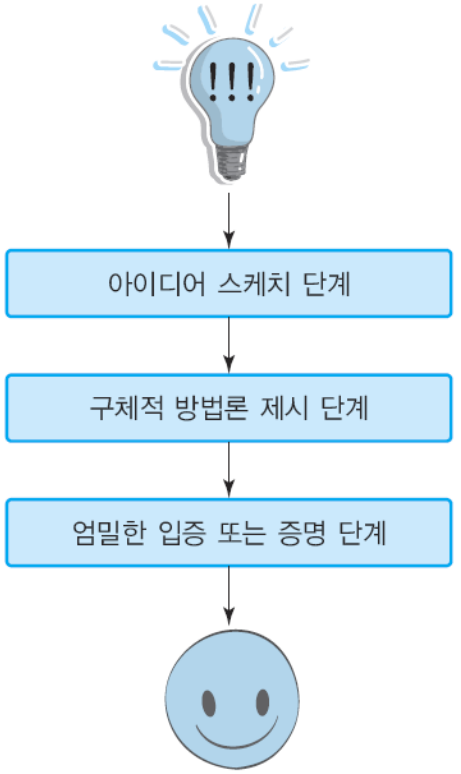
- 문제 해결의 핵심적인 실마리를 찾아내어 기술함
- 문제를 해결할 수 있는 방법론을 구상하게 되며 개략적인 아이디어를 스케치함

2. 구체적인 방법론 제시 단계

- 아이디어를 묶어서 구체적인 블록 다이어그램(block diagram) 등으로 표현함
- 프로그래밍의 경우 유사 코드(pseudo code) 단계까지 구체화하는 단계

3. 엄밀한 입증이나 증명의 단계

자기가 내린 결론을 객관적인 증명 방법을 통해 누구나 공감할 수 있게 증명함



〈그림 4.1〉 증명의 단계적 접근 방법

4.2 여러 가지 증명 방법

수학이나 공학에서의 증명 문제는 $p \rightarrow q$와 같은 논리 함축을 증명함

논리 함축 $p \rightarrow q$가 참이 되기 위해서는 p, q가 모두 참이거나 q에 관계없이 p가 거짓임을 보이면 됨

✓ 증명 방법은 직접 증명법과 간접 증명법 그리고 기타 증명법으로 구분함

- 직접 증명법은 $p \rightarrow q$를 직접 증명하는 것임
- 간접 증명법은 논리적 동치를 이용하거나 다른 특수한 방법으로 증명함

✓ 주어진 문제 유형에 따라 다양한 방법으로 접근하는 것이 효율적임

4.2 여러 가지 증명 방법


지금부터 7가지 증명 방법을
살펴봅시다.
수학적 귀납법
모순 증명법
직접 증명법
대우 증명법
존재 증명법
반례 증명법
필요충분조건 증명법

4.2.1 수학적 귀납법

수학이나 공학에서 새로운 결과를 얻는 2가지 중요한 방법론

- **연역법(deduction)**

  주어진 사실(facts)들과 공리(axioms)들에 입각하여 추론(inference)을 통하여 새로운 사실을 도출하는 것임

- **귀납법(induction)**

  관찰과 실험에 기반한 가설을 귀납 추론을 통하여 일반적인 규칙을 입증하는 것임

여기서 잠깐!!

수학적 귀납법을 최초로 사용한 사람은 16세기 이탈리아의 과학자 마우로리코(Maurolico)로 알려져 있다. 17세기 페르마(Fermat)와 파스칼(Pascal)이 이 방법을 사용했는데, 페르마는 이 방법을 '무한하강의 방법(method of infinite descent)'이라고 불렀다. 1883년 드 모르간(De Morgan)이 이런 과정을 신중히 기술하여 수학적 귀납법이라고 이름을 붙였다. 수학적 귀납법에 의한 증명은 현재 일반적으로 누구나 인정하는 공리(axiom)로 여겨지며 다양한 문제의 증명에 이용되고 있다.

수학적 귀납법(Mathematical Induction)

- 명제 p_1, p_2, p_3, …, p_n이 사실이라고 할 때, p_{n+1}의 경우에도 성립함을 보이면 됨
- 먼저 n이 1인 경우에 성립하는 것을 보이고, 모든 양의 정수 n에 대해 성립한다고 가정하면 n + 1의 경우에도 성립함을 보여주면 됨

  - **기초 단계(basic)**

    출발점이 되는 n의 값
  - **귀납 가정(inductive assumption)**

    p_1, p_2, p_3, …, p_n이 성립한다고 가정하면
  - **귀납 단계(inductive step)**

    p_{n+1}의 경우에도 성립함

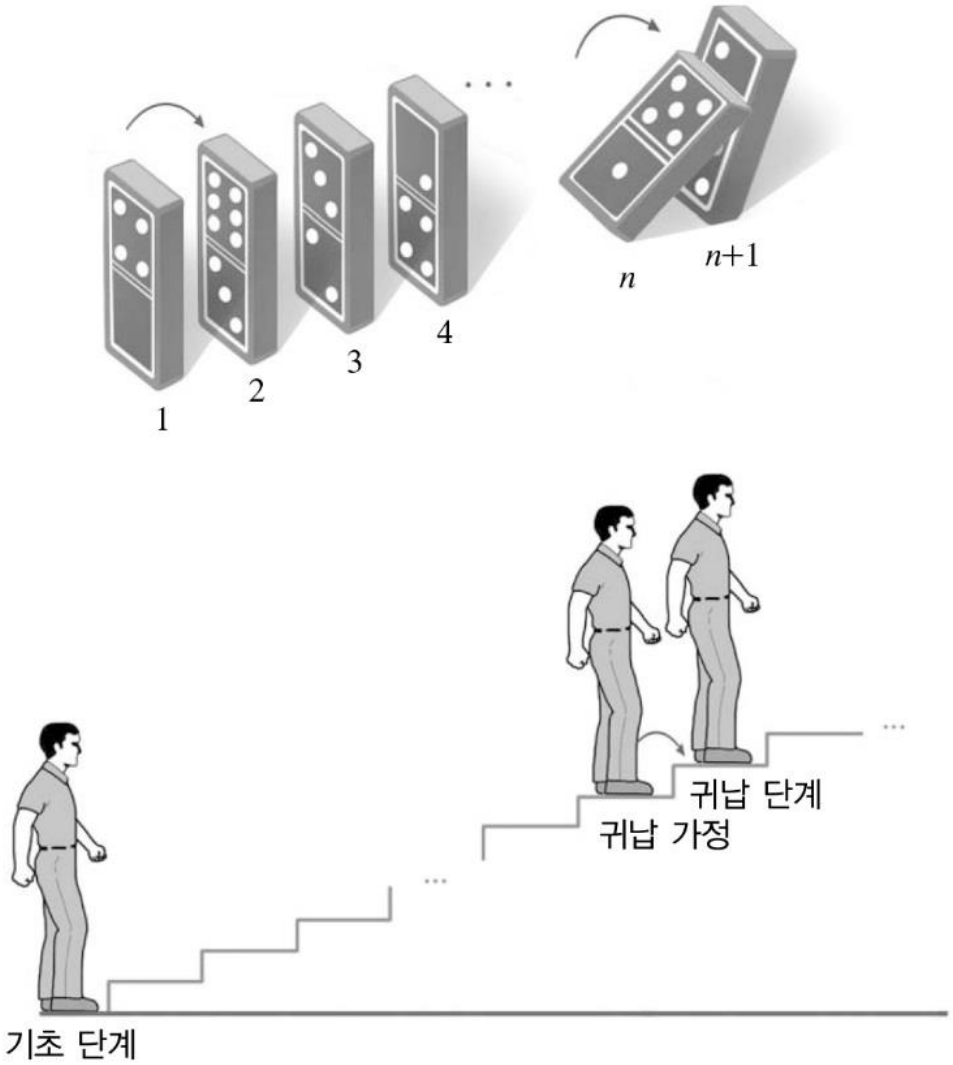
4.2.1 수학적 귀납법

수학적 귀납법을 보다 직관적이면서도 쉽게 이해하기 위해 도미노와 계단이 인용됨

예1) 첫 번째 도미노를 건드리면 두 번째 도미노가 넘어지고, 그 뒤에 있던 도미노 들이 연속해서 계속 넘어지면, n번째 도미노가 넘어지면 n + 1번째 도미노도 연속해서 넘어지는 현상 으로 비유됨

예2) 계단을 오를 경우, 첫 번째 계단을 오르고 그 후 n번째 계단을 지나 n+1번째 계단을 같은 방법으로 오르는 것에 비유됨

4.2.1 수학적 귀납법



〈그림 4.2〉 도미노와 계단 오르기

4.2.1 수학적 귀납법

수학적 귀납법은 논리식을 증명하는 데에도 이용

예) ‘ 모든 양의 정수 x에 대해 $p(x)$가 만족된다’ 는 명제를 증명하자

증명: $p(x)$에 모든 양의 정수 x를 대입한 경우, 즉 $p(1)$, $p(2)$, …, $p(n)$, $p(n+1)$이 모두 참(true)이 됨을 보임으로써 주어진 논리식을 증명함

정리 4-1

수학적 귀납법의 원리

모든 정수 n에 대해 어떤 명제 $p(n)$이 주어졌을 경우 $p(n)$이 $n \geq 1$인 모든 정수에 대해 참이라는 것을 증명하기 위한 방법은 다음과 같다.

① (기초 단계) $p(1)$이 참임을 보인다.

② (귀납 가정) $p(n)$이 참이라고 가정한다.

③ (귀납 단계) 귀납 가정에 입각하여 $p(n+1)$이 참임을 보인다.

4.2.1 수학적 귀납법

예제 ④-1 $S_n = \sum_{i=1}^{n} i = \frac{n(n+1)}{2}$임을 증명해보자.

증명 (n에 대한 수학적 귀납법을 이용)

(기초 단계) $n = 1$인 경우 왼쪽 $= S_1 = 1 = \frac{1 \cdot 2}{2} =$ 오른쪽

(귀납 가정) 만약 $S_n = \sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ 이라고 가정하면

(귀납 단계) 왼쪽 $= S_{n+1} = S_n + (n+1)$

$$= \frac{n(n+1)}{2} + (n+1)$$

$$= \frac{n^2 + n + 2n + 2}{2}$$

$$= \frac{n^2 + 3n + 2}{2}$$

$$= \frac{(n+1)(n+2)}{2}$$

$=$ 오른쪽 [n 대신 $n+1$을 각각 대입한 값]

∴ 위 의 식 이 성립

예제 ④-2 $1^2+2^2+\cdots+n^2=\dfrac{n(n+1)(2n+1)}{6}$임을 증명해보자.

증명 (n에 대한 수학적 귀납법을 이용)

(기초 단계) $n=1$인 경우 왼쪽 $=1^2=\dfrac{1\cdot 2\cdot 3}{6}=$ 오른쪽

(귀납 가정) 만약 $1^2+2^2+\cdots+n^2=\dfrac{n(n+1)(2n+1)}{6}$이라고 가정하면

(귀납 단계) 왼쪽 $=1^2+2^2+\cdots+n^2+(n+1)^2$

$$=\frac{n(n+1)(2n+1)}{6}+(n+1)^2$$

$$=\frac{n(n+1)(2n+1)+6(n+1)^2}{6}$$

$$=\frac{(n+1)\{n(2n+1)+6(n+1)\}}{6}$$

$$=\frac{(n+1)(2n^2+7n+6)}{6}$$

$$=\frac{(n+1)(n+2)(2n+3)}{6}$$

$$=\frac{(n+1)(n+2)\{2(n+1)+1\}}{6}$$

= 오른쪽 [n 대신 $n+1$을 각각 대입한 값]

∴ 위 의 식이 성립

4.2.1 수학적 귀납법

예제 ④-3 높이가 h인 포화 이진 트리(full binary tree)에서는 최대한 2^h개의 잎(leaf) 노드(node)를 가진다는 것을 증명해보자.

증명 (높이 h에 대한 수학적 귀납법을 이용)

(기초 단계) 높이 h가 0인 경우 잎 노드는 루트(root) 노드 하나뿐이다.

즉, $2^h = 2^0 = 1$

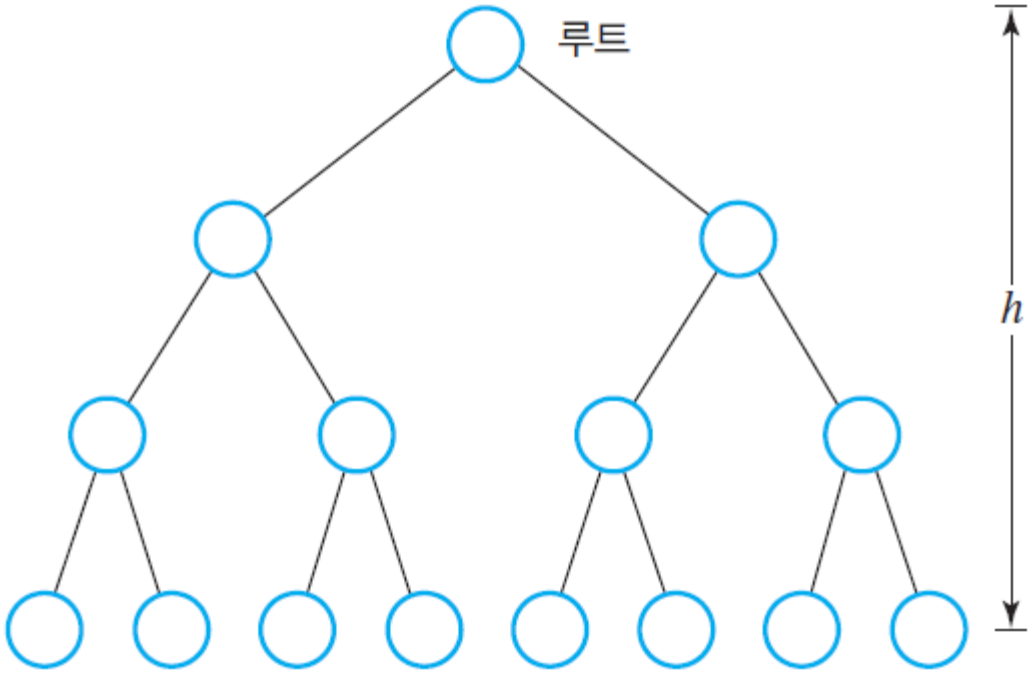
(귀납 가정) 만약 높이가 h인 경우 2^h개의 잎 노드를 가진다고 가정하자.

(귀납 단계) 높이가 $h+1$인 경우에는 2^h개의 잎 노드가 각각 2개씩의 잎 노드들을 가지므로 잎 노드의 총 개수는 $2^h \cdot 2 = 2^{h+1}$

따라서 $h+1$의 경우에도 적용된다.

∴ 위의 식이 성립됨

4.2.1 수학적 귀납법



〈그림 4.3〉 포화 이진 트리

4.2.1 수학적 귀납법

예제 4-4 $n \geq 4$인 모든 정수에 대하여 $2^n \geq n^2$임을 증명해보자.

증명 (n에 대한 수학적 귀납법을 이용)

(기초 단계) $n \geq 4$이므로 n이 성립하는 가장 작은 수는 $n=4$이다.

좌변 $= 2^n = 2^4 = 16 = 4^2 = 16 =$ 우변

그러므로 좌변 $\geq$ 우변이다. 즉, $n=4$일 때 성립한다.

(귀납 가정) $n \geq 4$인 모든 정수에 대하여 $2^n \geq n^2$이 성립한다고 가정하자.

(귀납 단계) 여기서는 $2^{n+1} \geq (n+1)^2$일 경우에도 성립함을 보인다.

$2^{n+1} = 2 \times 2^n \geq 2 \times n^2 = n^2 + n^2$

그런데 $n \geq 4$에 대해서는 항상 $n^2 \geq 2n+1$인 성질을 이용하면

$2^{n+1} \geq n^2 + n^2 \geq n^2 + 2n + 1 = (n+1)^2$이다.

그러므로 $2^{n+1} \geq (n+1)^2$ [n 대신 $n+1$을 대입한 값]

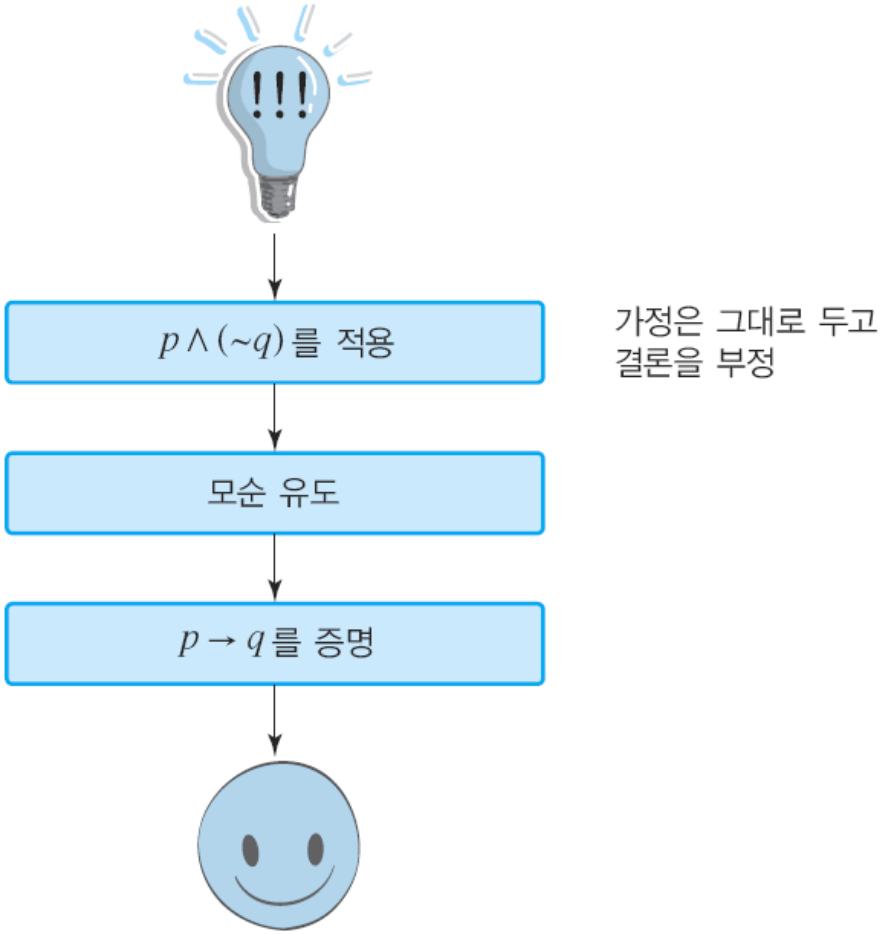
$\therefore$ 위의 식이 성립한다. ■

모순 증명법(Proof by Contradiction)

- 모순 증명법(또는 귀류법)은 기존의 전통적인 방법으로는 주어진 문제를 쉽게 증명할 수 없는 경우에 매우 유용함
- 일단 주어진 문제의 명제를 부정해 놓고 논리를 전개함
- 그 결과 모순됨을 보임으로써 본래의 명제가 사실임을 증명하는 방법임

- $P \rightarrow q$가 참인 것과 $p \wedge (\sim q)$가 거짓임은 동치이므로 $p \wedge (\sim q)$가 참이라고 가정하고, 그 결과 모순이 유도되면 원래의 명제가 참임을 증명한 셈임

4.2.2 모순 증명법



〈그림 4.4〉 모순 증명법

4.2.2 모순 증명법

예제 ④-5 'a, b가 실수일 때, $a+b-2>0$이면 $a>1$ 또는 $b>1$' 임을 증명해보자.

증명 먼저 결론을 부정하여 '$a\le 1$이고 $b\le 1$' 이라고 가정하면

$a-1\le 0$

$b-1\le 0$이 되므로

$a+b-2=(a-1)+(b-1)\le 0$가 된다.

즉, $a+b-2\le 0$

이 결과는 a, b가 실수일 때 $a+b-2>0$이라는 가정에 모순된다.

그러므로 원래의 명제 $a+b-2>0$이면 $a>1$ 또는 $b>1$이 성립한다.

예제 4-6 $\sqrt{2}$는 유리수(rational number)가 아님을 증명해보자.

$\sqrt{2}$가 유리수라고 가정하자. 그러면 유리수의 정의에 따라

$\sqrt{2} = \frac{n}{m}$ (n, m은 정수, $m \neq 0$, n, m은 서로소)으로 표현된다.

양변을 제곱해서 정리하면

$2m^2 = n^2$이 된다.

여기서 $2m^2$이 짝수가 되므로 n^2도 반드시 짝수여야 한다.

즉, n도 짝수이다.

따라서 $n = 2k$(k는 정수)로 표현될 수 있다. 이것을 위의 식에 대입하면

$2m^2 = 4k^2$

이므로

$m^2 = 2k^2$

그러므로 m^2이 짝수이고, 따라서 m도 짝수가 된다.

여기서 m과 n이 동시에 짝수가 되므로 n과 m이 서로소라는 가정에 모순된다.

따라서 $\sqrt{2}$는 유리수가 아니다. ■

4.2.2 모순 증명법

여기서 잠깐!!

유리수는 분수로 표현 가능한 실수이다. 즉, $\frac{b}{a}(a \neq 0)$로 나타낼 수 있는 실수이다.

$$\text{수} \begin{cases} \text{실수} \begin{cases} \text{유리수} \\ \text{무리수} \end{cases} \\ \text{허수} \end{cases}$$

예제 4-7 $1+3\sqrt{2}$가 무리수임을 증명해보자.

증명 (모순 증명법에 의한 증명)

$1+3\sqrt{2}$가 무리수가 아닌 유리수라고 가정하자.

그러면 유리수의 정의에 따라

어떤 정수 m과 n에 대해$(m \neq 0)$, $1+3\sqrt{2}=\frac{n}{m}$으로 표현된다.

따라서

$$3\sqrt{2}=\frac{n}{m}-1$$

$$=\frac{n-m}{m}$$

그러므로

$\sqrt{2}=\frac{n-m}{3m}$이 된다.

그러나 $n-m$과 $3m$은 정수이고, $m \neq 0$이므로

$\sqrt{2}$는 두 정수 $n-m$과 $3m$으로 표현되는 분수, 즉 유리수가 된다.

그러나 $\sqrt{2}$는 실제로 무리수이므로 모순이다.

그러므로 $1+3\sqrt{2}$는 무리수이다. ■

4.2.2 모순 증명법

예제 4-8 n이 자연수이고 n이 2가 아닌 소수(prime number)일 경우, n은 반드시 홀수가 됨을 증명해보자.

증명 (모순 증명법에 의한 증명)

'n이 2가 아닌 소수일 경우 $q(n)$은 n이 홀수이다' 라는 명제를 부정하여

'n이 2가 아닌 소수이고 또한 n은 짝수이다' 라고 가정한다.

n이 짝수이므로 $n = 2m$으로 표현될 수 있다. (m은 임의의 자연수)

자연수 m이 1인 경우에는 $n=2$이며,

$m > 1$이면 n은 m으로 나누어지므로 소수가 될 수 없다.

따라서 모순이다.

그러므로 n은 반드시 홀수가 된다. ■

직접 증명법(Direct Proof)

- 통상 주어진 유용한 정보로부터 추론을 통하여 목적하는 결론에 도달할 수 있도록 유도하는 증명법임
- 명제 $p \rightarrow q$의 직접 증명은 논리적으로 p의 진리 값이 참일 때 q도 참임을 보이는 증명 방법임

예제 4-9 만약 $6x + 9y = 7$이라면 x 또는 y가 정수가 아님을 증명해보자.

증명 먼저 $6x + 9y = 7$이라고 가정하자.
이것은 $3(2x + 3y) = 7$로 바꿀 수 있다.
즉, $2x + 3y = \frac{7}{3}$이 된다.
그러나 $\frac{7}{3}$은 정수가 아니므로 $2x + 3y$도 정수가 될 수 없다.
따라서 x 또는 y는 정수가 아니다. ■

4.2.3 직접 증명법

예제 ④-10 $|a| > |b|$일 때 $a^2 > b^2$임을 증명해보자.

증명 $a, b > 0$이고 $a > b$일 경우 우리는 $a^2 > b^2$임을 알고 있다.

그런데 어떤 a, b에 대해서도 $|a|, |b| > 0$이므로

$|a| > |b|$일 때 $|a^2| > |b^2|$이 된다.

이 경우 $|a^2| = a^2$이고 $|b^2| = b^2$이므로

$|a| > |b|$일 때 $a^2 > b^2$이 된다. ■

4.2.3 직접 증명법

예제 4-11 두 짝수의 합은 항상 짝수가 됨을 증명해보자.

증명 a와 b를 모두 임의의 짝수라고 하자.

짝수의 정의에 따라 임의의 정수 m과 n에 대해,

$a = 2m$

$b = 2n$

으로 나타낼 수 있다. a와 b를 합하면

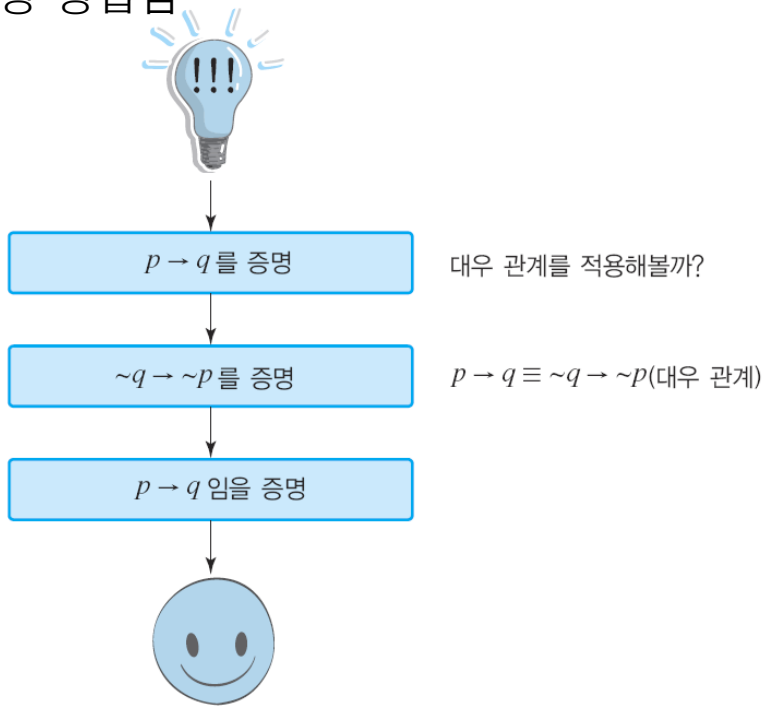
$$a + b = 2m + 2n$$
$$= 2(m + n)$$

$m + n$은 정수이기 때문에

$a + b = 2(m + n)$은 항상 어떤 정수값의 2의 배수이므로 짝수가 된다. ■

대우 증명법(Contrapositive Proof)

- $p \rightarrow q$와 $\sim q \rightarrow \sim p$가 대우 관계로서 논리적 동치가 됨을 이용하여, $\sim q \rightarrow \sim p$가 참인 것을 증명임
- $p \rightarrow q$가 참이 되는 것을 논리적 동치 관계를 이용하여 간접적으로 보여주는 증명 방법임



〈그림 4.5〉 대우 증명법

4.2.4 대우 증명법

예제 4-12 x가 짝수이면 x는 2이거나 소수가 아님을 증명해보자.

증명 대우 증명법에 따라 $\sim q \rightarrow \sim p$를 적용하면
'x가 2가 아니고 소수이면 x는 홀수이다' 를 증명하면 된다.
그런데 x가 2가 아닌 소수는 모두 홀수이므로
원래의 명제가 성립한다. ■

4.2.4 대우 증명법

여기서 잠깐!!

완전수(perfect number)는 그의 약수(divisor)들 중 자신을 제외한 모든 약수들의 합과 같은 수이다. 예를 들면, 6의 약수는 1, 2, 3, 6인데 자기 자신을 제외하면 1, 2, 3이다. 1+2+3=6이므로 6은 완전수가 된다.

예제 4-13 완전수는 소수가 아님을 증명해보자.

증명 주어진 명제의 대우는 '소수는 완전수가 아니다' 이므로 이 대우 명제가 참인 것을 보이면 된다.

p를 소수라고 가정하면 정의에 따라 $p \geq 2$이고 p의 약수는 1과 p뿐이다. 이때 p를 제외한 약수는 1밖에 없으므로 자신을 제외한 모든 약수들의 합은 1이다.

따라서 p는 완전수가 아니다.

그러므로 완전수는 소수가 아니다. ■

4.2.4 대우 증명법

예제 4-14 모든 정수 n에 대해 n^2이 짝수라고 가정하면 n도 짝수임을 증명해보자.

증명 주어진 명제의 대우인 '만약 n이 홀수이면 n^2은 홀수이다'를 증명한다.

n이 임의의 홀수라고 가정하면 홀수의 정의에 따라 어떤 정수 k에 대해 $n = 2k + 1$로 표현될 수 있다.

양변을 제곱해서 계산하면

$$n^2 = (2k+1)^2$$
$$= 4k^2 + 4k + 1$$
$$= 2(2k^2 + 2k) + 1\text{이다.}$$

정수의 곱과 합은 정수이므로 괄호 안에 있는 $2k^2 + 2k$도 당연히 정수가 된다.

그러므로 $n^2 = 2 \cdot (2k^2 + 2k) + 1$이므로 n^2은 홀수이다. ■

4.2.4 대우 증명법

예제 ④-15 n이 자연수이고 n이 2가 아닌 소수라면 n이 홀수임을 증명해보자.

증명 $p \rightarrow q$의 증명을 대우인 $\sim q \rightarrow \sim p$로 증명해보자.

주어진 명제의 대우는 'n이 짝수이면 n=2이거나 n은 소수가 아니다'이다.

n이 짝수라고 가정하면 $p < n$인 어떤 자연수 p에 대하여 $n = 2 \cdot p$가 된다.

이때 $p = 1$이거나 $p > 1$이다.

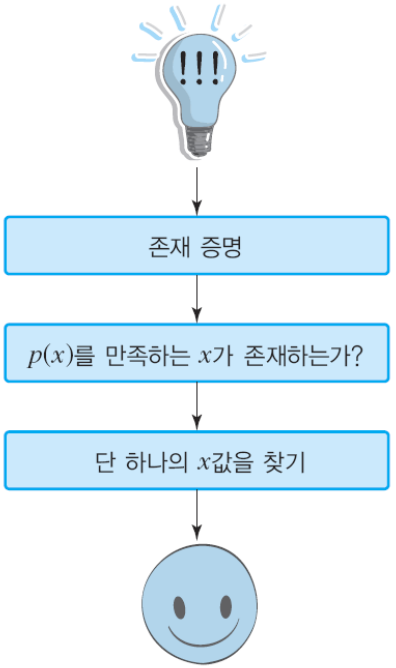
(1) $p = 1$이면 $n = 2$이다.

(2) $p > 1$이면 n은 p로 나누어지므로 n은 소수가 아니다.

그러므로 주어진 명제는 참이 된다. ■

존재 증명법(Existence Proof)

- $p(x)$를 x라는 변수를 가지는 명제라고 한다면
- $p(x)$가 참인 x가 적어도 하나가 존재한다는 것을 보이는 증명 방법임
- '$\exists x$ such that $p(x)$'를 보이는 것임



〈그림 4.6〉 존재 증명법

4.2.5 존재 증명법

예제 4-16 $p(x)$가 술어 'x는 정수이고 $x^2=289$' 일 때 이 식을 만족하는 x가 존재함을 증명해보자.

증명 제곱근을 구하는 방법을 사용하여 $p(x)$를 만족하는 x의 존재 여부를 결정할 수 있다. 이 경우 $x = 17$일 때 이 식이 만족함을 보인다. ■

예제 4-17 a가 0이 아닌 실수이고 b가 실수일 때, 방정식 $ax + b = 0$을 만족시키는 실수 x가 존재함을 증명해보자.

증명 가정에서 $a \neq 0$이므로
방정식의 해를 구하는 방법에 따라
x는 $-\frac{b}{a}$이며 이는 실수이다.
따라서 주어진 명제가 참이다. ■

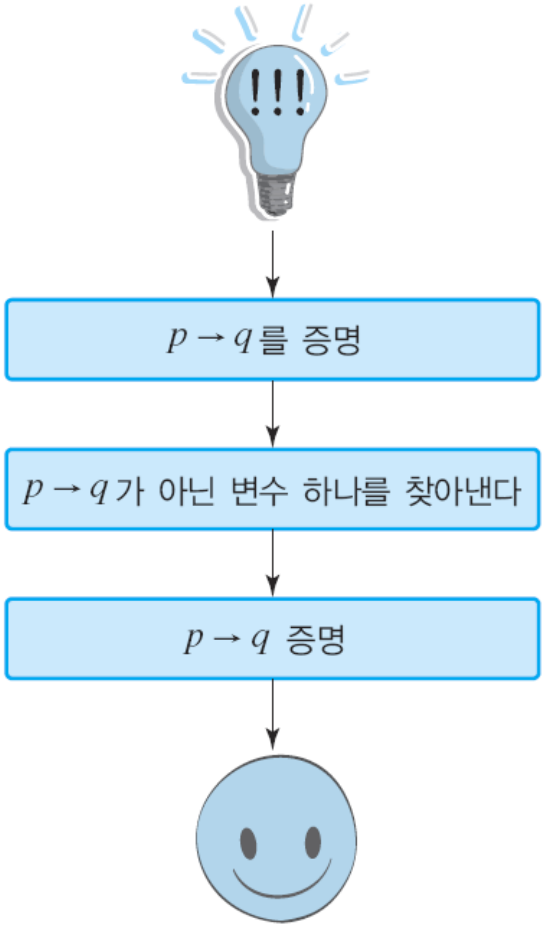
반례 증명법(Proof by Counter-example)

- 어떤 명제가 참 또는 거짓임을 입증하기 어려운 경우에 효과적인 증명 방법임
- 주어진 명제에서 모순이 되는 간단한 하나의 예를 보임으로써 비교적 쉽게 증명할 수 있는 방법임

  ✓ $\forall x\, p(x)$이 거짓임을 보이기 위해 $\sim[\forall x\, p(x)]$와 동치인 $\exists x \sim p(x)$에서 $p(x)$를 만족하지 않는 x가 적어도 하나 존재함을 보임

  ✓ 이 경우 x를 반례(counter-example)라고 함

4.2.6 반례 증명법



〈그림 4.7〉 반례 증명법

4.2.6 반례 증명법

예제 4-18 'p가 양의 정수이고 $x = p^2 + 1$이면 x는 소수이다' 란 명제가 거짓임을 증명해 보자.

증명 위의 명제가 거짓임을 입증하기 위해서는 어떤 양의 정수 p에 대해 x가 소수가 아닌 예를 하나만 보이면 된다.
가령 $p = 3$일 경우 $x = 10$이므로 x는 소수가 아니다.
따라서 주어진 명제는 거짓이다. ■

예제 4-19 모든 실수 x에 대해 $(x+1)^2 \geq x^2$이 성립하지 않음을 증명해보자.

증명 반례를 들어서 위의 명제가 거짓임을 증명한다.
가령 $x = -1$일 때 $(-1+1)^2 = 0 < 1 = (-1)^2$이다.
따라서 위의 명제가 성립하지 않는다. ■

4.2.6 반례 증명법

예제 4-20 '모든 실수 a와 b에 대하여 $a^2=b^2$이면 $a=b$이다' 가 거짓임을 증명해보자.

증명 이 문제의 경우에는 2가지 방법으로 증명할 수 있다.

(1) 직접 증명법으로 하면

$a^2=b^2$이면 $a^2-b^2=0$

$(a-b)(a+b)=0$

따라서 $a=b$ 또는 $a=-b$

그러므로 $a=b$인 결론은 거짓이라는 것을 증명할 수 있다.

(2) 그러나 이런 경우에는 반례의 예를 들어 증명하는 것이 매우 편리하다.

가령 $a=1$, $b=-1$이라고 가정한다면

$a^2=1^2=1$이고 $b^2=(-1)^2=1$이다.

따라서 $a^2=b^2$을 만족하지만

$1\neq -1$이므로 $a\neq b$이다.

그러므로 주어진 명제는 거짓이다. ■

필요충분조건 증명법(if and only if proof)

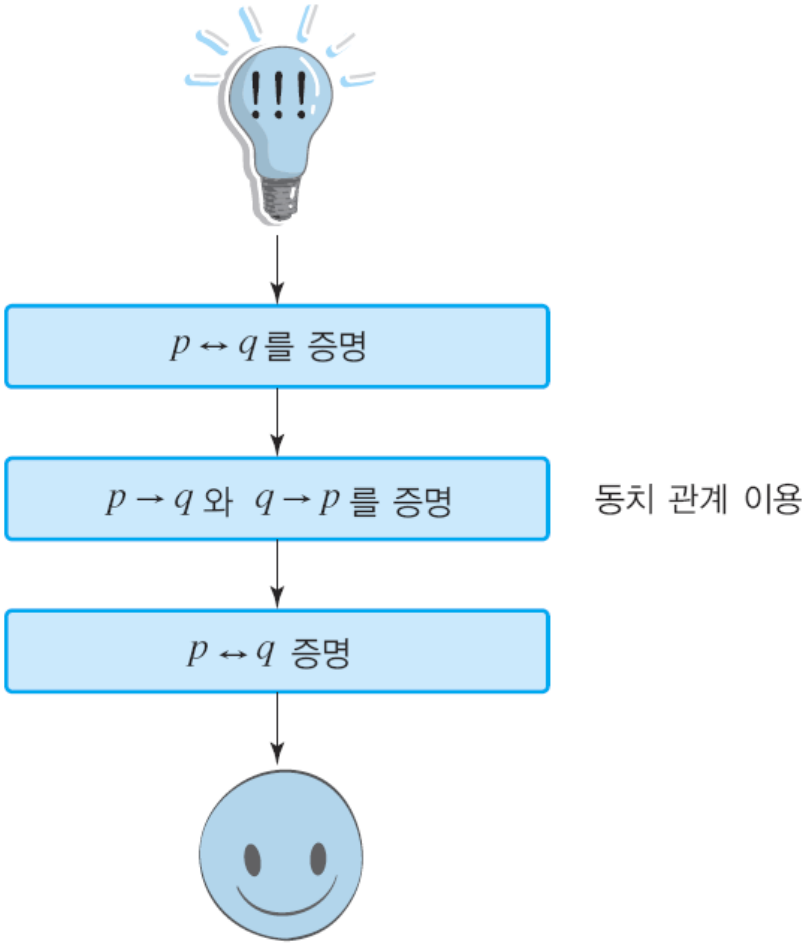
- 주어진 명제의 동치를 통하여 증명함
- 'p if and only if q' 를 증명하기 위해 '만약 p이면 q이다' 와 '만약 q이면 p이다' 의 두 가지를 증명함

P의 필요충분조건 $(p \leftrightarrow q)$를 보이기 위해

$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$ 이므로

$p \rightarrow q$와 $q \rightarrow p$를 통해 증명

4.2.7 필요충분조건 증명법



〈그림 4.8〉 필요충분조건 증명법

4.2.7 필요충분조건 증명법

예제 4-21 모든 정수 n에 대해, $n-1$이 짝수임과 n이 홀수임이 동치라는 것을 증명해 보자.

증명 이 증명에서는 모든 정수 n에 대해 'n이 홀수이면 $n-1$은 짝수이다'와 '$n-1$이 짝수이면 n은 홀수이다'와 같이 두 가지 경우로 나누어서 증명한다.

(1) n이 홀수이면 $n-1$은 짝수이다.

만약 n이 홀수이면 어떤 정수 k에 대해 $n=2k+1$로 나타낼 수 있다.

$$n-1=(2k+1)-1=2k$$

가 된다. 그러므로 $n-1$은 짝수이다.

(2) $n-1$이 짝수이면 n은 홀수이다.

만약 $n-1$이 짝수이면 어떤 정수 k에 대하여 $n-1=2k$로 나타낼 수 있다.

이 경우에

$$n=2k+1$$

이 된다. 따라서 n은 홀수이다.

위의 2가지 경우가 모두 성립하므로 동치이다. ■

4.3 프로그램의 입증

- 증명 못지않게 엄밀한 정확성이 요구되는 컴퓨터 프로그램을 입증하는 것 또한 매우 중요함
- 정확한 프로그램이 되기 위해서는 구문 오류(syntax error)를 포함하지 않아야 함
- 예상치 못하게 끝나서도 안 되며, 주어진 입력에 대해 정확한 결과를 도출해야 함
- 프로그램의 정확성에 대한 입증이 필요함

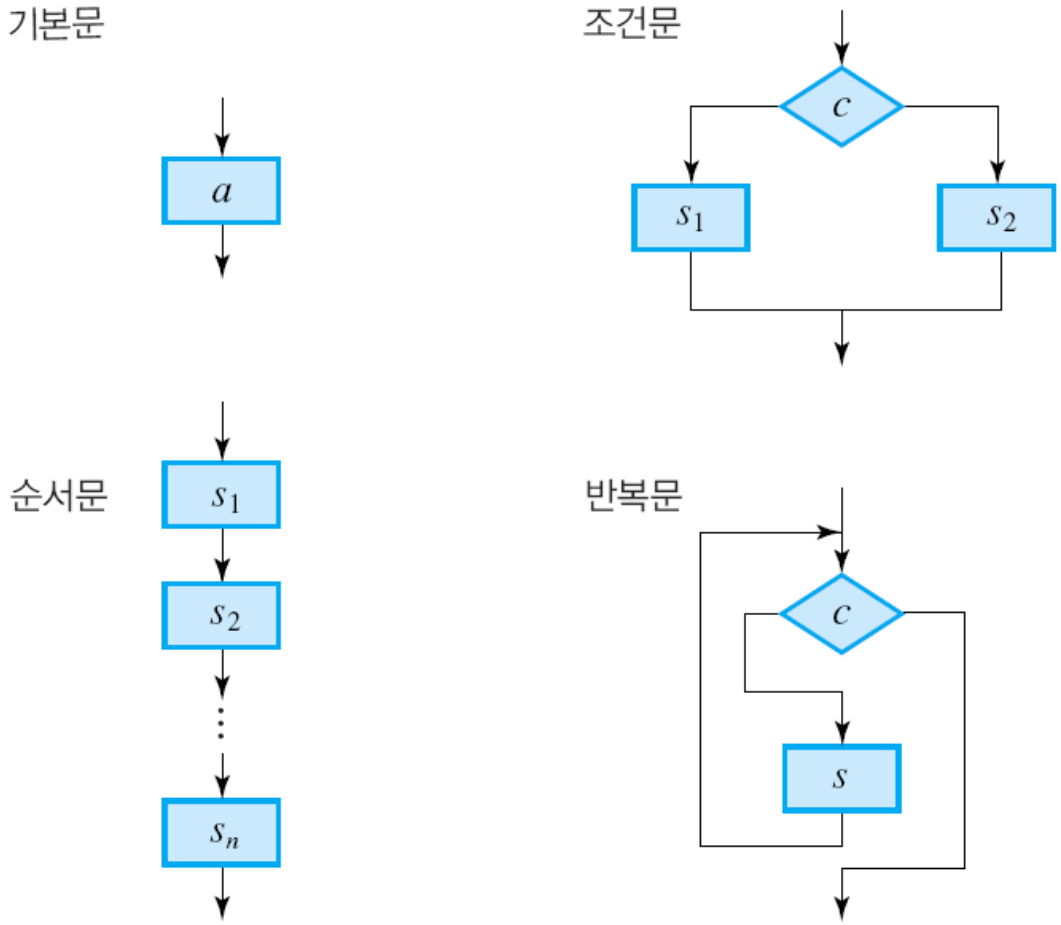
미국의 우주항공센터(NASA)에서는 인공위성을 쏘아 올리기 전에 유능한 프로그래머들이 작성한 소프트웨어를 수학자들이 수학적으로 입증한다. 이것은 프로그램의 입증을 통하여 어떠한 오류도 방지하기 위함임

〈그림 4.9〉 인공위성 발사 장면

프로그램의 입증

주로 제어 구조(control structure)에서 4가지의 문장 구조를 입증함

1. 순서문(Sequential statements)
2. 조건문(Conditional statements)
3. 반복문(Repeated statements)
4. 무조건적 이동문(Unconditional transfer statements)



〈그림 4.10〉 프로그램의 제어 구조

예제 4-22 다음과 같이 1부터 n까지의 정수값을 합하는 프로그램에서 정확성을 입증해 보자.

```
void Sum_n()
{
    int i, n, sum = 0;
    scanf("%d", &n);
    for (i = 1;  i <=n;i++)
        sum += i;
    printf("%d, %d", n, sum);
}
```

풀이 sum의 초기값이 0이고 for 문장을 한 번 수행하면 i = 1의 값이 sum에 더해져서 sum = 1이 된다. 여기서 for 문장을 한 번 더 수행하면 sum = 1+2가 된다. 이와 같은 방법으로 n번을 수행할 경우에는

$$\text{sum} = 1$$

$$\text{sum} = 1 + 2$$

$$\text{sum} = 1 + 2 + 3$$

$$\text{sum} = 1 + 2 + 3 + \cdots\cdots + (n - 1)$$

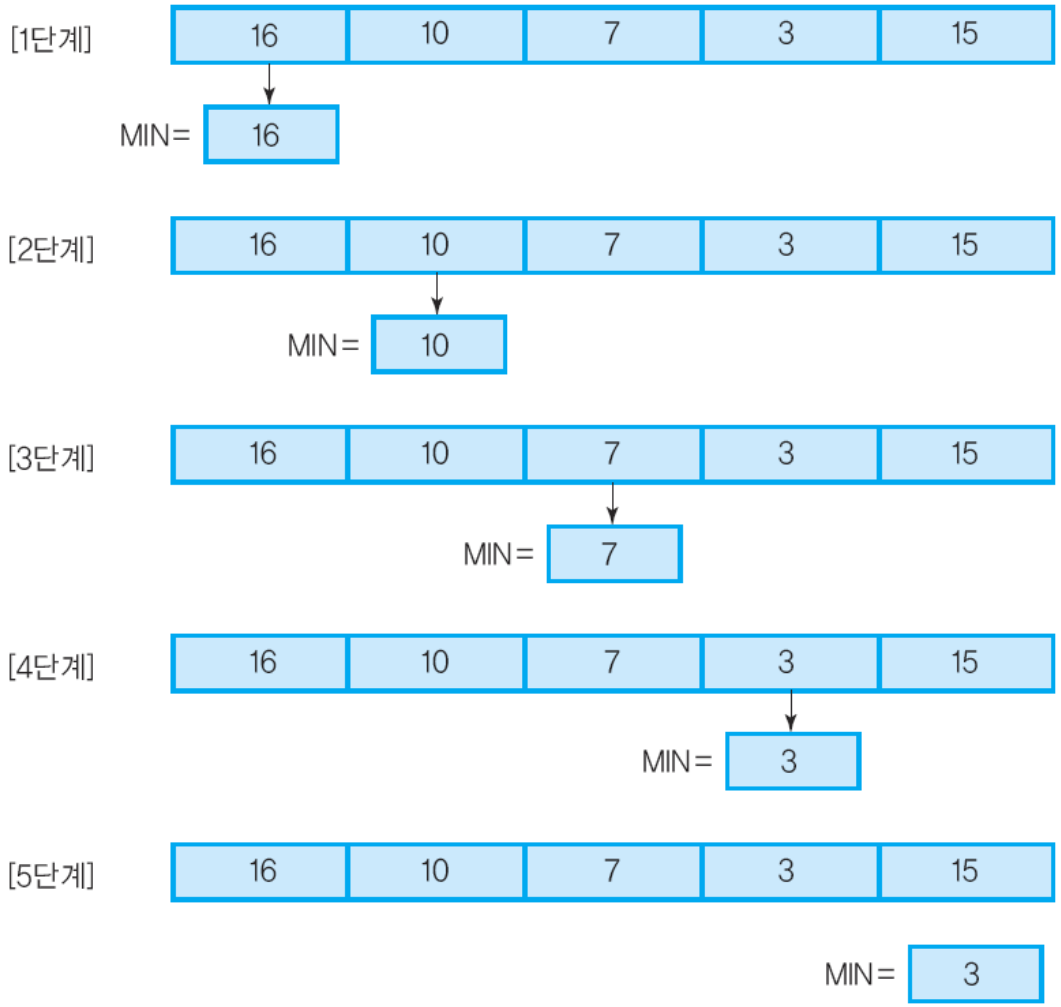
$$\text{sum} = 1 + 2 + 3 + \cdots\cdots + (n - 1) + n$$

이 된다. 따라서 이 프로그램은 수학적 귀납법의 방식과 같이 1부터 n까지의 합을 정확하게 구하는 것이 입증된다. ■

예제 4-23 n개의 양의 정수 중에서 가장 작은 수를 찾는 다음과 같은 프로그램을 입증해 보자.

```
Find_Min (int array[ ], int MIN)
{
   int i ;
   MIN = array[0];
   for (i = 1; i < n; i++)
      if (MIN > array[i]) MIN = array[i];
   return (MIN);
}
```

풀이 이 프로그램에서 array[ ]는 n개의 정수를 저장하는 배열로 선언되었다. 먼저 배열의 첫 번째 값이 MIN에 저장된다. 계속해서 for 문장에서 배열의 값을 읽어서 MIN보다 작을 경우에는 그 값이 MIN에 주어진다. 그 결과 for 문장에서 그때까지 읽은 값들 중 가장 작은 값이 MIN에 저장되는 셈이다. 따라서 n번의 과정을 거친 후에는 MIN에 있는 값이 전체 배열 중에서 가장 작은 값이라는 것이 입증된다. ■



〈그림 4.11〉 MIN을 구하는 과정

증명에서의 오류

증명은 미리 알려진 사실과 증명된 결과, 또는 이전에 증명된 문장으로 이루어진다. 이때 세심한 주의를 기울이지 않는다면 증명에 있어서 오류가 일어날 수 있다. 예를 들어, 다음과 같은 증명을 생각해보자.

$$
\begin{aligned}
1 &= \sqrt{1} \\
&= \sqrt{(-1)\cdot(-1)} \\
&= \sqrt{(-1)}\cdot\sqrt{(-1)} \\
&= \left(\sqrt{(-1)}\right)^2 \\
&= -1
\end{aligned}
$$

여기서 우리는 $1 = -1$이라는 것을 보여주었는데 이런 과정은 거짓이다. 그러면 오류는 어디에서 일어났을까? 이 증명에서 $\sqrt{(-1)\cdot(-1)} = \sqrt{(-1)}\cdot\sqrt{(-$ 의 계산에서 일어났다.

우리는 실수 a, b에 대하여 $\sqrt{ab} = \sqrt{a}\sqrt{b}$ 가 성립하는 것을 사용하려 했으나 a와 b가 음이 아닌 실수일 때만 성립한다는 점을 간과한 것이다. 따라서 우리는 증명에 있어서 매우 세심한 주의를 기울여야 한다는 것을 알 수 있다.

다음과 같은 지혜로운 말을 증명이나 입증할 수 있을까요?

- 성경에 보면 "누구든지 죄가 없는 사람은 이 여인을 돌로 쳐라"는 말에 누구도 감히 나서지 못했답니다.
- 셰익스피어의 명작 《베니스의 상인》에서 빚을 갚지 않으면 살 한 근을 베어내기로 했지만, "정확히 한 근을 베어내지 못하면 너는 큰 처벌을 받을 것이다"라는 말에 감히 칼을 대지 못했답니다.
- 조선 선조 임금 때의 백사 이항복은 어릴 적 어머니가 출타하면서 참깨 한 가마니의 개수를 세어놓으라고 했는데 어린 항복은 한 홉의 개수만을 잠깐 동안에 세어놓고 나가서 놀았답니다. 한 홉의 10배가 한 되가 되고, 한 되가 10개 모여서 한 말이 되고, 말이 모여서 한 가마니가 되는 법을 이용한 것이지요.
- 지난 여름 강릉 경포대에 놀러 갔다가 경포해수욕장의 모래의 개수를 세어보았더니 모두 10의 30승 개 정도였어요. 다른 사람들에게 그 개수를 이야기했더니 모두들 "에이~ 엉터리!"라고 말하더군요. 그러나 아무도 그것에 대해 합리적으로 반박하지는 못했어요. 왜 그럴까요?

요약

- 증명이란 논리적 법칙을 이용하여 주어진 가정으로부터 결론을 유도해내는 추론의 한 방법으로서 어떠한 명제나 논증이 적절하고 타당한지를 입증하는 작업이다.
- 공학이나 컴퓨터 관련 학문에 있어서 주어진 문제를 해결하기 위해서는 단계적 접근이 매우 효과적이다. 첫째, 아이디어 스케치 단계이다. 둘째, 구체적인 방법론 제시 단계이다. 셋째, 엄밀한 입증이나 증명의 단계이다
- 연역법은 주어진 사실들과 공리들에 입각하여 추론을 통하여 새로운 사실을 도출하는 것이며, 귀납법은 관찰과 실험에 기반한 가설을 귀납 추론을 통하여 일반적인 규칙을 입증하는 것이다.
- 증명 방법에는 수학적 귀납법, 모순 증명법, 직접 증명법, 대우 증명법, 존재 증명법, 반례 증명법, 필요충분조건 증명법 등이 있다.
- 수학적 귀납법에 의한 증명에서는 명제 p_1, p_2, p_3, ⋯, p_n이 사실이라고 할 때 p_{n+1}의 경우에도 성립한다는 것을 보이게 된다.

요약

- 모순 증명법은 주어진 문제의 명제를 일단 부정해놓고 논리를 전개하여, 그것이 모순됨을 보임으로써 본래의 명제가 사실임을 증명하는 방법이다.
- 직접 증명법은 명제 $p \rightarrow q$를 직접 증명하는 것으로, p의 진리값이 참일 때 논리적으로 q도 참임을 보이는 것이다.
- 대우 증명법은 $p \rightarrow q$와 $\sim q \rightarrow \sim p$가 논리적으로 동치임을 이용하여, $\sim q \rightarrow \sim p$가 참인 것을 증명함으로써 $p \rightarrow q$가 참인 것을 보여주는 방법이다.
- 존재 증명법은 $p(x)$를 x라는 변수를 가지는 명제라고 할 때 $p(x)$가 참인 x가 적어도 하나가 존재한다는 것을 보이는 증명 방법이다. 즉 '$\exists$x such that $p(x)$'를 보이는 것이다.
- 반례 증명법은 주어진 명제에서 모순이 되는 하나의 예를 간단히 보임으로써 비교적 쉽게 증명할 수 있는 증명법이다.

요약

- 필요충분조건 증명법은 동치를 통하여 증명하는데, 'p if and only if q' 를 증명하기 위해서 '만약 p이면 q이다' 와 '만약 q이면 p이다' 를 증명한다.
- 프로그램의 입증은 주로 제어 구조에서 필요한데 순서문, 조건문, 반복문, 무조건적 이동문 등의 문장 구조를 입증한다.

응용

- 증명법의 응용
  - 수학
  - 자연과학
  - 여러 가지 공학 등에 폭넓게 활용되며 학문적 기반을 이루는 기초가 됨