220914 Computer Thinking

1. 서론 - 프로그래밍과 논리 / 수학

- 프로그래밍의 어려운 점 두 가지
  - 프로그래밍 언어 문법과 라이브러리 사용
  - 논리 (Hard Logic)

[참고]

- 명제
  - 참이나 거짓을 알 수 있는 식이나 문장
  - p, q, r, ...로 표현
- 진릿값
  - 참이나 거짓을 표현
  - T, F 또는 1, 0

[연산 (결합)]

- 부정 NOT
  - p가 명제일 때, 명제의 진릿값이 반대
  - ~p 또는 p로 표기 (not p또는 p의 부정으로 읽음)

`p`	`~p`
T	F
F	T

- 논리곱 AND
  - p, q가 명제일 때, p, q 모두 참일 때만 참이 되는 명제
  - $p \wedge q$ (p and q, p 그리고 q)

`p`	`q`	`p∧q`
T	T	T
T	F	F
F	T	F

p	q	p∧q
T	F	F

- 논리합 OR
  - p, q가 명제일 때, p, q 모두 거짓일 때만 거짓이 되는 명제
  - $p \vee q$ (p or q, p 또는 q)

p	q	pvq
T	T	T
T	F	T
F	T	T
F	F	F

- 배타적 논리합 XOR
  - p, q가 명제일 때, p, q 중 하나만 참일 때 참이 되는 명제
  - $p \oplus q$ (p xor q)

p	q	p⊕q
T	T	F
T	F	T
F	T	T
F	F	F

[합성]

- 연산자 우선 순위
- 항진명제 : 진릿값이 항상 참
- 모순명제 : 진릿값이 항상 거짓
- 사건명제 : 항진명제도 모순명제도 아닌 명제
- 조건명제
  - p, q가 명제일 때, 명제 p가 조건(또는 원인), q가 결론(또는 결과)로 제시되는 명제
  - $p \rightarrow q$ (p이면 q이다)

p	q	p → q

p	q	p → q
T	T	T
T	F	F
F	T	T
F	F	T

- P가 F면 p → q 는 T
- Q가 T면 p → q 는 T

- 쌍방조건명제
  - p, q가 명제일 때, 명제 p와 q가 모두 조건이면서 결론인 명제
  - p ↔ q (p이면 q고, q이면 p다)

p	q	p ⇔ q
T	T	T
T	F	F
F	T	F
F	F	T

- 조건명제의 역, 이, 대우
  - 역 : q → p
  - 이 : ¬p → ¬q
  - 대우 : ¬q → ¬p

p	q	p → q	q → p	¬p → ¬q	¬q → ¬p
T	T	T	T	T	T
T	F	F	T	T	F
F	T	T	F	F	T
F	F	T	T	T	T

- 문제 4
  - `p ∧ (q → ~p)`

p	q	~p	(q → ~p)	p ∧ (q → ~p)

`p`	`q`	`~p`	`(q → ~p)`	`p ∧ (q → ~p)`
T	T	F	F	F
T	F	F	T	T
F	T	T	T	F
F	F	T	T	F

[증명]

- 증명은 정확한 명제식으로 표현할 수 있는 것이라야 함
- 보통은 정확한 명제식까지 쓰지는 않으나 근본적으로는 명제식으로 바꿀 수 있음
- 증명에 대한 수많은 오해가 p → q를 p ↔ q와 혼동하는 것에서 일어남
- 수학적 귀납법과 증명의 수준
  - 수학적 귀납법의 기본형 : P(1)이 참이고, P(n) → P(n+1)이 참이면 P(n)이 모든 자연수 n에 대해서참이다
  - 수학적 귀납법의 강한 형태

증명 연습

- Trivial Proof : ∀x,P(x) → Q(x)를 증명하려는데, Q(x)가 항상 참인 경우
  - 문제1. 다음 명제를 증명하시오
    -
    -
- Vacuous Proof : ∀x,P(x) → Q(x)를 증명하려는데, P(x)가 항상 참인 경우
  - 문제 2. 다음 명제를 증명사시오

1. 논리와 증명

- 문제 1: 다음 명제들이 항진명제라는 것을 진리표를 이용해서 보이시오

① $\sim(\sim p \wedge q) \vee q$

② $(\sim p \vee q) \vee (p \wedge \sim q)$

p	q	~p	$(\sim p \wedge q)$	$\sim(\sim p \wedge q)$	$\sim(\sim p \wedge q) \vee q$
T	T	F	F	T	T
T	F	F	F	T	T
F	T	T	T	F	T
F	F	T	F	T	T

- 문제 2: 다음 명제들이 모순명제라는 것을 진리표를 이용해서 보이시오

① $(\sim p \vee q) \wedge (p \wedge \sim q)$

② $(p \wedge q) \wedge (p \wedge \sim q)$

p	q	~p	$(\sim p \vee q)$	~q	$(p \wedge \sim q)$	$(\sim \boldsymbol{p} \vee \boldsymbol{q}) \wedge (\boldsymbol{p} \wedge \sim \boldsymbol{q})$
T	T	F	T	F	F	F
T	F	F	F	T	T	F
F	T	T	T	F	F	F
F	F	T	T	T	F	F

- 문제 3: 다음 명제의 쌍 들에 대해서 두 명제가 동등한지를 진리표를 이용해 확인하시오

① $p \wedge (p \vee q)$와 p

② $\sim p \vee \sim q$와 $\sim(p \vee q)$

p	q	$(p \vee q)$	$\boldsymbol{p} \wedge (\boldsymbol{p} \vee \boldsymbol{q})$
T	T	T	T
T	F	T	T
F	T	T	F
F	F	F	F

- 문제 4: 명제식의 변형을 통하여 다음 명제를 간소화하시오.

① $(p \wedge \sim q) \vee (p \wedge q)$ $= p \wedge (\sim q \vee q)$

u

$p \wedge u \Rightarrow$

T

p	u	
T	T	T
F	T	F

- 문제 5: 다음 명제들이 참인지 확인하시오. 단, R은 실수의 집합을 의미하고, Z는 정수의 집합을 의미한다.

① $\forall x \in R, x^2 \geq x$ – F. 반례를 찾을 수 있으면... $x = \frac{1}{2}$

② $\forall x \in Z, x^2 \geq x$

③ $\exists x \in R, x^2 < x$

④ $\exists x \in Z, x^2 < x$

- 문제 5: 다음 명제들이 참인지 확인하시오. 단, R은 실수의 집합을 의미하고, Z는 정수의 집합을 의미한다.

① $\forall x \in R, x^2 \geq x$

② $\forall x \in Z, x^2 \geq x$

③ $\exists x \in R, x^2 < x$ $\frac{1}{2}$

④ $\exists x \in Z, x^2 < x$

- 문제 6: (직접 증명) n이 짝수이면 $3n + 5$는 홀수임을 증명하라.

(힌트: $n = 2k$로 두고 $3n + 5$가 2(어떤 정수) $+ 1$ 형태로 표현될 수 있는지··

2의 배수

$2k$ $\quad 3 \cdot 2k + 5$

$4 + 1$

$\underbrace{2(3k+2)}_{\text{2의 배수}} + \underbrace{1}_{+1}$

- 문제 7: n이 홀수이면 n^2+n은 짝수임을 증명하라.

2k+1

- 문제 8: m이 짝수이고 n이 홀수이면 $2m+3n$은 홀수임을 증명하라

2k　　2l+1　　2의배수+1

$2\cdot 2k+3(2l+1)$

$=2\cdot 2k+2\cdot 3l+\underbrace{3}_{2+1}$

$=2(2k+3l+1)+1$

- 문제 9: (대우를 증명) 자연수 n에 대해, n^2+5가 홀수이면 n은 짝수임을 증명하라

(힌트: 명제 대신, n이 홀수이면 n^2+5은 짝수임을 증명한다)

$n^2+5=2k+1$　　$(2k+1)^2$

- 문제 10: n^2이 짝수이면 n은 짝수임을 증명하라.

(대우) n이 홀수면 n^2은 홀수

$n=2k+1$

$(2k+1)^2=4k^2+4k+1$

$=2(2k^2+2k)+1$ 홀수.

대우가 참 ⇒ 명제는 참

- 문제 12: n^2이 3의 배수이면 n은 3의 배수임을 증명하라.

대우를 이용해야하고 나머지가 1인경우 나머지가 2인경우

3k+1, 3k+2

2. 수와 표현

- 컴퓨터는 0과1을 표현할 수 있는 비트들을 모아 수를 표현
- k개의 비트를 사용하면 0부터 2^k^-1까지 표현 가능
- - 어떤 값 n을 표현하기 위해서는 몇 개의 비트가 필요할까?
  - $2^k - 1 \geq n$이 성립해야 함 -> 즉, $2^k \geq n + 1$
  - 같은 의미로, $k \geq \log(n+1)$ -> 약 $\log n$ 비트가 필요
  - $x = \log n$과 $2^x = n$은 같은 말

  $2^k = n$

  $k = \log_2 n$

  $\log n$

  k 비트..

  $0 \sim 2^k - 1 \geq n$ (10)

  $k=3$ 7.

  $=4$ 15
- log n이란
  - - 위의 식을 잘 보면, logn이란

    $\boxed{2^k = n \quad k = \log n}$

    (가) 2의 몇 승이 n이 되느냐의 답

    (나) n을 표현하는 데 몇 비트가 필요한가의 답

    (다) 1로 시작해서 계속 두 배를 할 때 몇 번 하면 n이 되느냐의 답

    (라) n을 2로 계속 나눌 때 몇 번 나누면 거의 1이 되느냐에 대한 답
- x = log n 일때 x 와 n을 비교하여
- 32비트 컴퓨터의 주소 공간은 2^32^ = 약 40억개 주소

2.1 문제

■ 문제들

k 비트

- 문제 1: 2진수 표현에서 logn 비트로 표현할 수 있는 숫자 범위는?

n

2^k

$2^{\log n}$ $\quad n^{\log 2^1} = n$

- 문제 3: n이 충분히 큰 값일 때 다음 중 어느 값이 더 큰가? 각 쌍에 대해 비교하고 그 이유를 작성하시오.

① $2n$ ( ) n^2

② $2^{\frac{n}{2}}$ ( ) $\sqrt{3^n}$

③ $2^{n \log n}$ $= n^n$ (>) $n!$ $\quad 2^{n \log n} = 2^{\log(n^n)} = n^{n \log 2^1} = n^n$

④ $\log 2^{2n}$ ( ) $n\sqrt{n}$

$2^k = n, \quad k = \log n.$

문제 5: 다음 함수들의 역함수를 구하시오.

① $f(x) = \log(x-3) - 5$

② $f(x) = 3\log(x+3) + 1$

③ $f(x) = 2 \times 3^x - 1$

$\boxed{f(x)+5} = \log(x-3)$

$2^{f(x)+5} = x - 3$

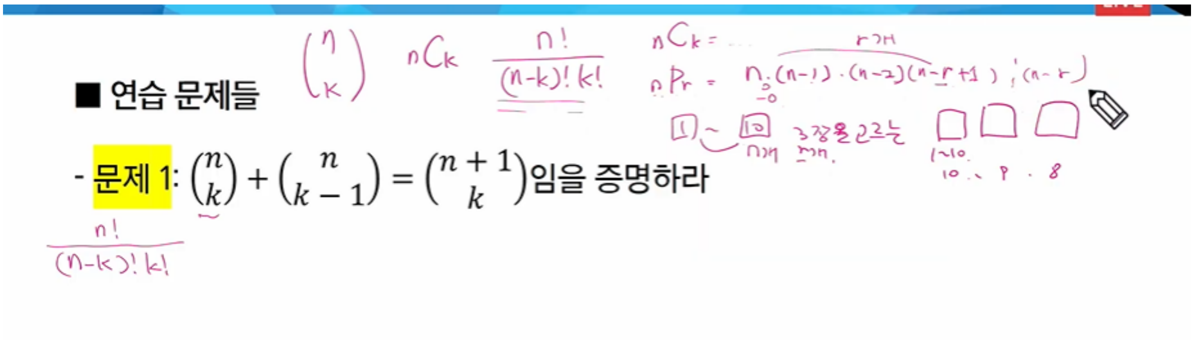
$x = 2^{f(x)+5} + 3$

$f^{-1}(x) = 2^{x+5} + 3$

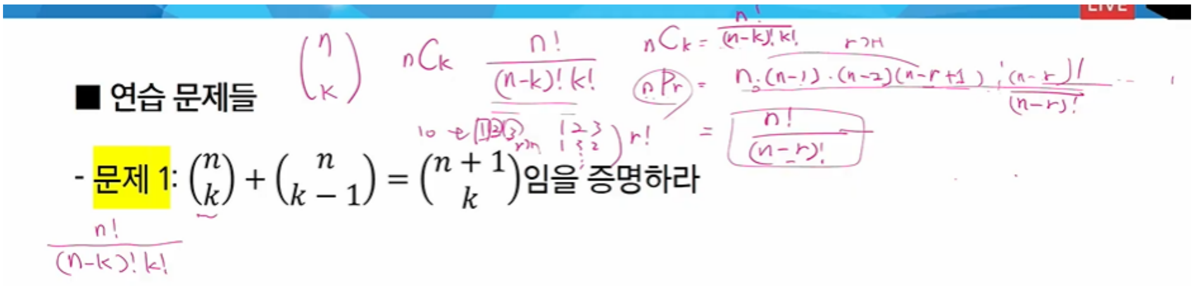
3. 집합과 조합론

- 두 집합 A와 B에 대해 A가 B의 부분집합임을 증명한다는 것은
- 조합론은 경우의 수를 따지는 문제들을 보통 말한다
- 조합은 개수는 C를 이용하여 표현하기도 하지만
- $_5C_2$

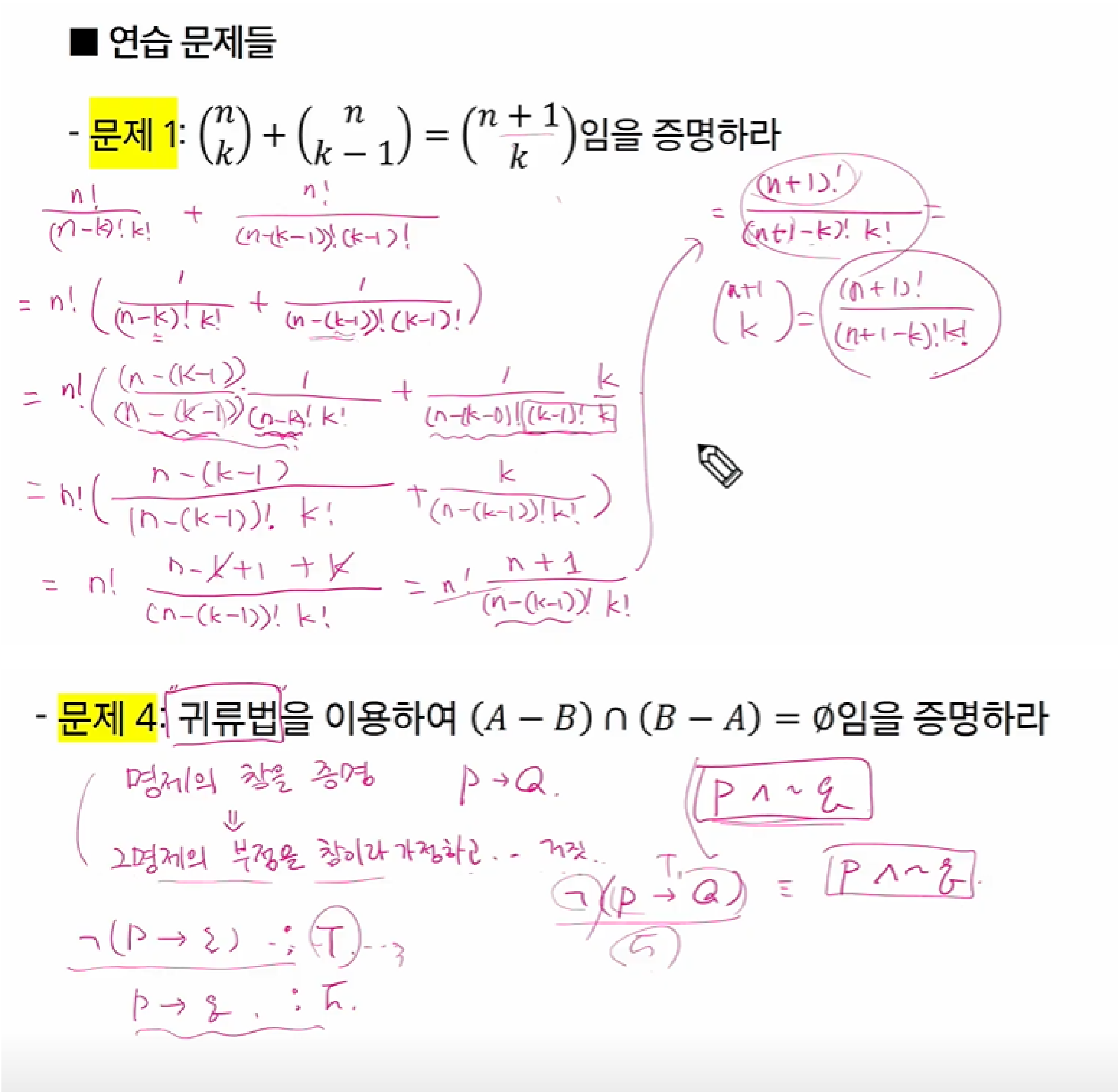
3.1 연습문제



■ 연습 문제들
- 문제 1: $\binom{n}{k}+\binom{n}{k-1}=\binom{n+1}{k}$임을 증명하라



■ 연습 문제들
- 문제 1: $\binom{n}{k}+\binom{n}{k-1}=\binom{n+1}{k}$임을 증명하라



■ 연습 문제들
- 문제 1: $\binom{n}{k}+\binom{n}{k-1}=\binom{n+1}{k}$임을 증명하라
- 문제 4: 귀류법을 이용하여 $(A-B)\cap(B-A)=\emptyset$임을 증명하라
명제의 참을 증명
그 명제의 부정을 참이라 가정하고 .. 거짓

- **문제 4**: 귀류법을 이용하여 $(A-B)\cap(B-A)=\emptyset$임을 증명하라

$p \in (A-B)\cap(B-A)$, $\boxed{p\in(A-B)}$, $\boxed{p\in(B-A)}$

1. $p \in A$, $\boxed{p\notin B}$: $(B-A)\subset B$
   $\boxed{p \notin (B-A)}$
2. $p\notin A$, $p\in B$ $\quad (A-B)\subset A$.
   $p\notin A-B$
3. $p\notin A$, $p\notin B$.
4. $p\in A$, $p\in B$
   $p\in A\cap B$. $\quad p\notin(A-B)$
   $\quad p\notin(B-A)$

- **문제 8**: 8 × 8 체스 판에 말 두개를 놓으려고 한다. 아무 곳에나 놓아도 되지만 한 칸에 두개가 들어가지는 못한다. 가능한 방법은 모두 몇가지인가?

$${}_{64}C_2 = \frac{64!}{62!\,2!} = \frac{\overset{32}{\cancel{64}}\cdot 63}{\cancel{2}\cdot 1} = 2016 \text{ 가지..}$$

$$\binom{64}{2}$$

$$\begin{array}{r} 63 \\ \times\ 32 \\ \hline 126 \\ 189 \\ \hline 2016 \end{array}$$

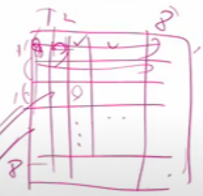
- **문제 8**: 8 × 8 체스 판에 말 두개를 놓으려고 한다. 아무 곳에나 놓아도 되지만 한 칸에 두개가 들어가지는 못한다. 가능한 방법은 모두 몇가지인가?

0 A [0 … i … j … N] A의 원소중 2개를 고르는 방법...

```
for i : 0 → N-2
    for j : i+1 → N-1
        f(i, j)
```

- 2016 가지..

for r : 1→



2개의 벽을 제거 했을때......

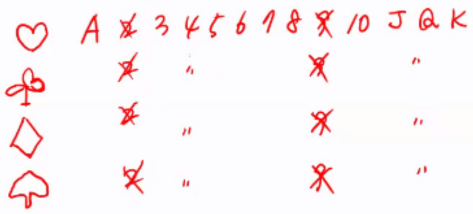
벽의 좌표를 저장한 (N) 배열.

A | (1,2) | (3,4) | ...... |

- **문제 12**: 52개의 카드를 이용해서 만들 수 있는 5개 카드의 조합은 몇가지인가?

$$\binom{52}{5} = \frac{52!}{(52-5)!\,5!} = 2,598,960$$

- 문제 16: 52개 카드에서 5개 카드 조합을 만들 때, 숫자가 같은 카드가 한 쌍도 없는 경우는 몇가지인가?



$_{52}C_1 \cdot {}_{48}C_1 \cdot {}_{44}C_1 \cdots$

1,317,888

- 순서가 있음

4. 기초 수식

- 알고리즘의 시간 복잡도를 표현할 수 있는 다양한 수식들이 존재
- 풀이법을 익혀 두어야 알고리즘의 시간 복잡도를 계산할 수 있고, 알고리즘이 시간이 얼마나 걸릴지 예측 가능

4.1 연습 문제

■ 연습 문제들: 다음 재귀식들을 O() notation 수준으로 풀어라.

- 문제 1: $T(n) = T(n-1) + 1$, T(0) = 1

```
f(n)
  if n ≤ 1
     return 1
  else
     return n * f(n-1)
              1
```

■ 연습 문제들: 다음 재귀식들을 O() notation 수준으로 풀어라.

- 문제 1: $T(n) = T(n-1) + 1$, T(0) = 1

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

$= T(n-3) + \boxed{1 + 1 + 1}$

$\vdots$

$= T(n-k) + k$

$= T(0) + n, \quad n = k.$

$= \boxed{1 + n}$

$O(n)$

- 문제 2: $T(n) = T(n-1) + n, T(0) = 1$

$O(n^2)$

- 문제 2: $T(n) = T(n-1) + n$, $T(0) = 1$

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

$T(n-2) = T(n-3) + n-2$

$\therefore O(n^2)$

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

$T(n) = T(n-3) + n-2 + n-1 + n$

$T(n) = T(n-k) + n-(k-1) + n-(k-2) + \cdots + n$

$n = k$,

$T(n) = T(0) + n-n+1 + n-n+2 + \cdots + n$

$= 1 + \underline{1 + 2 + \cdots + (n-1) + n}$

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