



2017-2학기

1장-계산적 사고란 무엇인가?

박종혁 교수 (컴퓨터공학과)

jhpark1@seoultech.ac.kr

<http://www.parkjonghyuk.net>

➤ 1장. 계산적 사고

1. 컴퓨터, 어디에나 있는 컴퓨터
2. 컴퓨터, 컴퓨터 과학, 그리고 계산적 사고
3. 주판에서 기계까지
4. 최초의 소프트웨어
5. 현대적인 컴퓨터란?
6. 최초의 현대적인 컴퓨터
7. 무어의 법칙
8. 참고문헌

➤ 조별과제

- 다음 주 강의 시간 발표

- 계산적 사고의 개념에 대한 실질적 정의를 제시한다.
- 데이터에 대한 아날로그 표현과 디지털 표현의 차이를 소개한다.
- 데이터의 표현, 저장, 처리를 위한 예로서 주판을 사용한 기계적인 계산의 유래를 알아본다.
- 현대적인 컴퓨터의 하드웨어와 소프트웨어의 발명에 기여한 역사적인 주요 사건들을 알아본다.
- 내장 프로그램 개념과 소프트웨어 실행 및 데이터 처리에 있어서의 그 역할을 설명한다.
- 현대적인 컴퓨터의 기본 구성과 특징을 소개한다.
- 무어의 법칙과 그 영향을 설명한다.

1.1 컴퓨터, 어디에나 있는 컴퓨터

- ❖ 소프트웨어 세상을 바꾼다 (동영상 -4m8s, 14m04s 시청)
https://www.youtube.com/watch?v=0U0ve_HFUL8
- ❖ 인간의 발명품 중에서 컴퓨터보다 더 많이 세상을 변화시킨 것이 있을까?
 - 회계, 공학기술, 작가, 학자, 기자, 그래픽 아티스트, 모션 픽쳐 애니매이터
 - 제정문제 관리, 전투장비, 소셜 네트워크
 - 사이버 컴퓨터 범죄
- ❖ 컴퓨터가 일상생활에 기치고 있는 무제한의 영향에 대해 실제로 우리는 어떠한 영향도 선택하지 못함
- ❖ 컴퓨터에 대해 공부하고 그것을 자신의 편의를 위해 사용하거나 컴퓨터기술을 거부하는 것 – 둘중 하나만 선택

❖ 컴퓨터(Computer)/컴퓨터 시스템(Computer System)

: 컴퓨터 하드웨어와 소프트웨어를 하나 언급

❖ 하드웨어(hardware)

: 데스크톱 또는 랩톱 컴퓨터를 구성하는 모든 물리적 장치

- 키보드, LCD, 메모리, 디스크드라이브, CD/DVD 드라이브, 마우스, CPU 등

❖ 소프트웨어(software)

: 컴퓨터 프로그램 통칭

❖ 프로그램 (program)

: 어떤 작업을 수행하는 명령어

❖ 코드 (code)

: 소프트웨어나 프로그램

- ❖ **앱 (app)/애플리케이션 (application)**
: 프로그램이 컴퓨터 하드웨어를 특정한 목적에 적용하는 수단이 되는 프로그램
- ❖ **프로그래머 (programmer)/소프트웨어 개발자 (software developer)**
: 프로그램을 만드는 일을 직업으로 가지는 사람
- ❖ **컴퓨터 과학 (computer science)**
: 컴퓨터에 관련된 모든 문제를 포함
 - 컴퓨터구조, 소프트웨어공학, 그래픽스, 로봇틱스, 정보보안, 네트워킹, 인공지능 등
- ❖ **계산적 사고 (computational thinking)**
: 컴퓨터 과학자들이 생각하고 추론하는 방식
 - 동영상(5m), <https://www.youtube.com/watch?v=OoRrKmmIEeY>

❖ 주판 : 초기의 계산 장치

- 저장소 : 숫자 값(데이터)을 저장하고 있음
- 표현 방법 : 메모리내 전자적 신호를 사용
- 계산 : 소프트웨어(소프트웨어는 컴퓨터에게 계산을 수행하도록 함)
- 사용자 인터페이스 : 인간이 기계와 소통하는 방식
 - 그래픽 사용자 인터페이스(GUI) : 그래픽 이미지를 조작

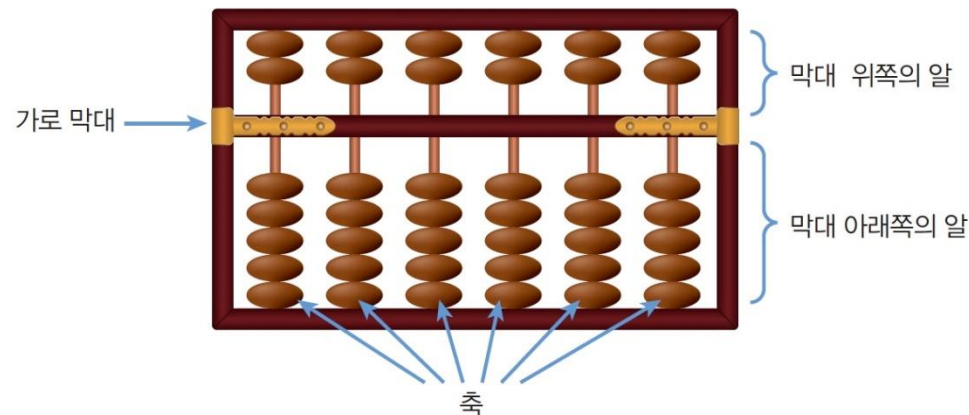


그림 1.2 중국식 주판

1.3 주판에서 기계까지

❖ 네이피어의 계산봉

- 막대마다 다른 방식으로 숫자가 위치.
막대 배열방식에 따라 곱하기, 나누기, 제곱근을 계산

❖ 파스칼린과 라이프니츠 계산기

- 회전반과 둥근 핸들을 돌림. 계산속도가 빠름. 사칙연산

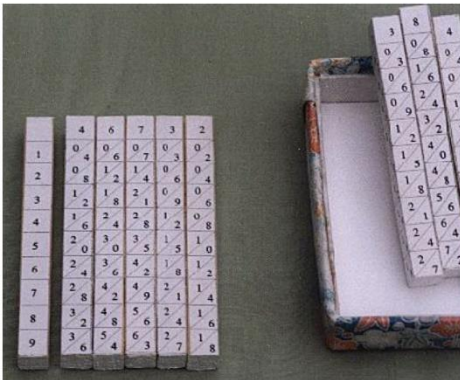


그림 1.5 네이피어의 계산봉(Napier's bones)

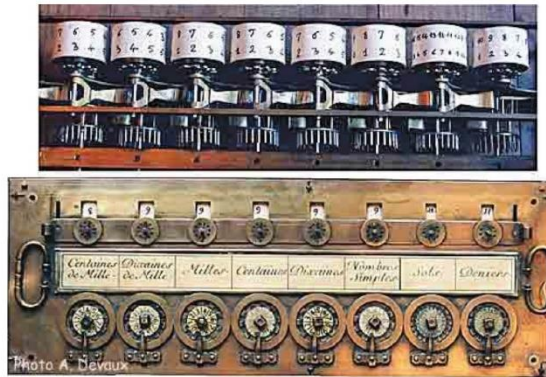


그림 1.6 파스칼린(Pascaline)

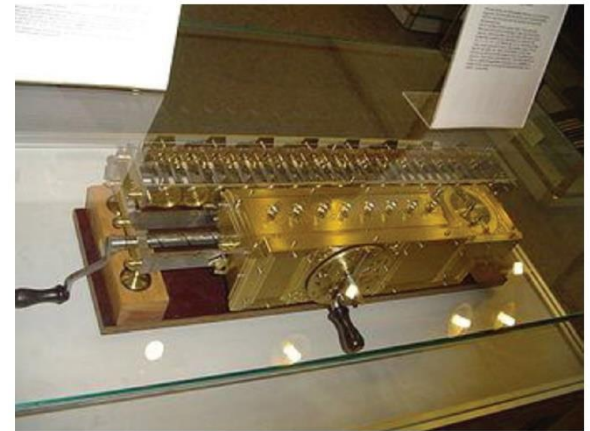


그림 1.7 라이프니츠 계산기(Leibniz' calculator)

❖ 안티키테라 메커니즘

- 계산을 위해서 톱니바퀴를 사용한 첫 번째 장치
- 해, 달, 행성의 위치 알아내기 위한 것



그림 1.8 안티키테라 메카니즘(Antikythera mechanism)의 잔해

- ❖ 프로그램은 장치에게 실행 방법을 명령
- ❖ 프로그램마다 다른 결과를 만들어 내야 함
- ❖ 자카드 직기 (Jacquard loom)
 - : 구멍이 뚫린 종이 카드 두루마기를 프로그램으로 사용
 - 프로그램이 가능한 첫 번째 기계
 - 천공카드의 구멍의 위치와 개수를 바꿈 으로서 다른 패턴의 천을 짤

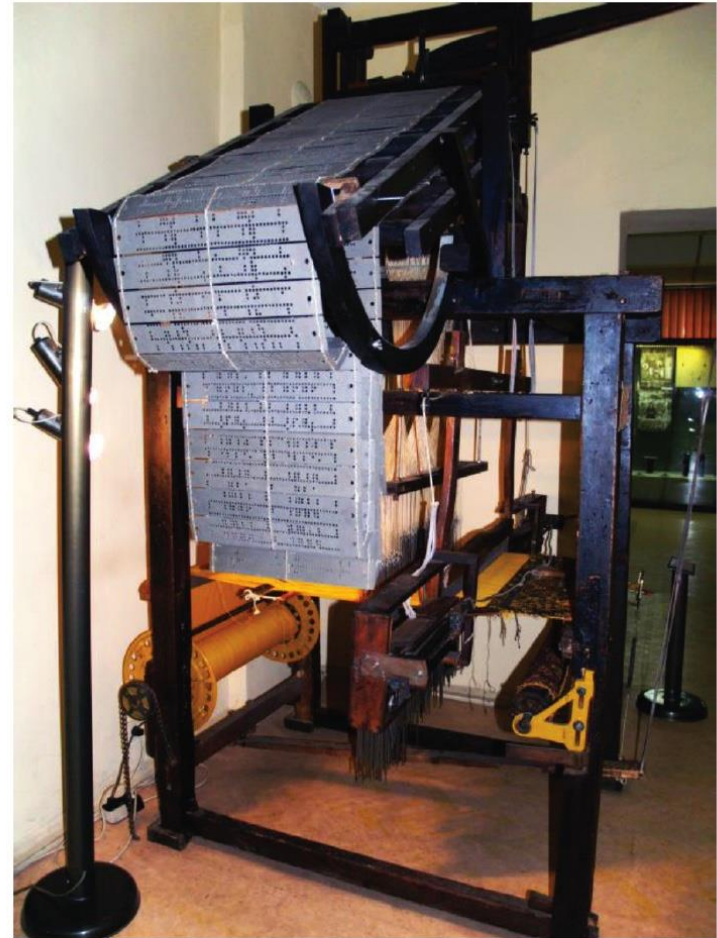
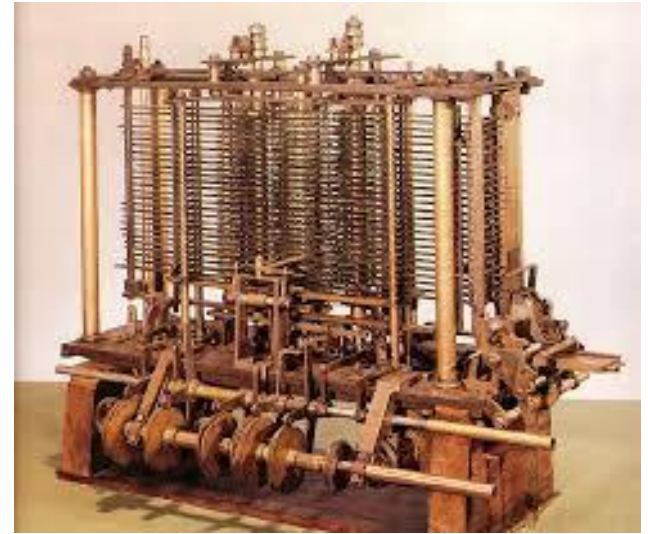


그림 1.9 자카드 직기의 모델

❖ 해석기관 (Analytical Engine)

- 찰스 베비지가 발명
- 천공카드 개념 도입
- 현대 컴퓨터에서 컴퓨터 프로그램이 지시하는 대로 복잡한 수학적 연산을 수행하는 것과 동일한 방법으로 일련의 수학적 연산을 수행



❖ 에이다 러브레이스 (Lovelace, Ada)

- 해석기관을 위한 프로그램을 작성
- 최초의 컴퓨터 프로그래머



1. 전용 기계식이 아닌 전자식 (electronic)

- 전기로 동작
- 허먼 홀러리스(Hollerith, Herman)의 미국 인구조사 도표 작성 기계

2. 아날로그가 아닌 디지털 (digital)

- 아날로그 (analog) : 연속적인 변화
 - 빛의 밝기, 전류량, 바람의 세기
- 디지털 (digital) : 무한한 개수의 가능성이 없고 변화도 연속적이지 않음
 - 9시30분과 9시31분

3. 내장 프로그램 개념 (stored program concept)을 가짐

- 현대 컴퓨터의 4가지 필수적인 부분
 - 입력 (input) 장치
 - 출력 (output) 장치
 - 메모리 (memory)
 - 프로세서 (processor)

❖ 입력장치

- 컴퓨터에 데이터를 입력
- 키보드, 컴퓨터 마우스, 트랙 패드, 마이크로폰, LCD

❖ 출력장치

- 컴퓨터가 자신의 계산결과를 사용자와 공유
- 컴퓨터 화면표시장치, 스피커와 프린터

❖ I/O

- 입력과 출력의 조합
- 디스크 드라이브, 플래쉬 메모리 카드, CD, DVD



그림 1.12 단순한 데스크톱 컴퓨터의 기본 구성

❖ 클릭 컴퓨터속으로 (4m26s)

<https://www.youtube.com/watch?v=K5khQ6yzMsU>

- ❖ **메모리 (memory)**
 - 입력되는 데이터 저장
- ❖ **프로세서 (processor)**
 - 입력을 **처리**(process)
 - 메모리와 프로세서는 작은 집적회로로 구성
 - **기계어** (machine instruction)를 처리
 - 프로세서마다 다른 기계어를 처리
 - 명령어를 실행과 메모리로부터 명령어를 가져오는것을 제어
- ❖ **내장 프로그램 개념 (stored program concept)**
 - 프로그램이 컴퓨터 메모리로 이동하여 적재되어야 프로세서가 프로그램을 사용
- ❖ **현재 컴퓨터 메모리의 목적**
 - 데이터를 저장
 - 데이터를 처리하는 명령어를 저장

❖ 미분 해석기 (differential analyzer)

- 미분방정식을 풀기 위해서 설계
- 전기를 사용
- 아날로그 회로로 구현

❖ ENIAC (Electronic Numerical Integrator and Computer, 1947)

- 존 모클리와 피터에커트에 의해 발명
- 최초의 전자식 컴퓨터 주장 특허출원 (1974)
 - 1973년 무효화됨
- 디지털
- 전기로 작동
- 저장된 프로그램 개념 사용
- 진공관(메모리 장치) 사용
- 계전기(기계식의 크기가 큰 전기적인 스위치) 사용



그림 1.14 진공관

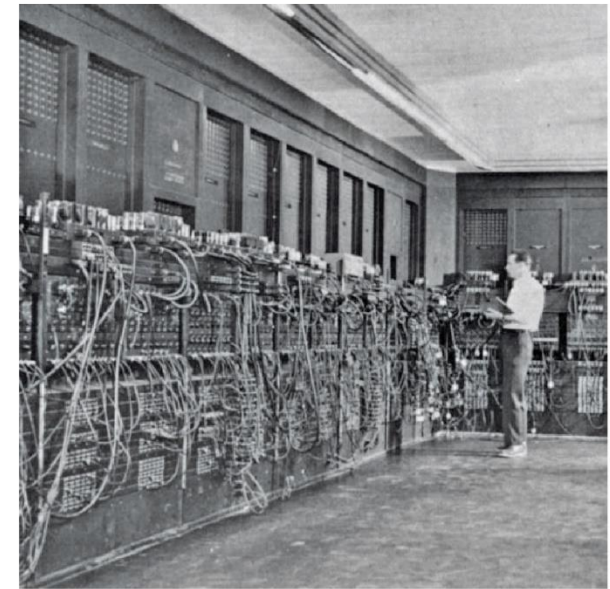


그림 1.13 에니악(ENIAC) - 초기의 현대적인 컴퓨터

❖ ABC 컴퓨터

- 존 아타나소프와 척 베리가 발명 (1937~38)
- 저장된 프로그램 개념을 사용하지 않았음
- 범용으로 프로그램이 가능하지 않음
- 실제로 현대적 컴퓨터로 인정받지 못함
- 3가지 중요한 측면에서 최초로 인식
 - 프로그램 가능한 최초의 완전 전자식 계산기
 - 전자식 메모리를 최초로 반영
 - 이진수를 최초로 사용

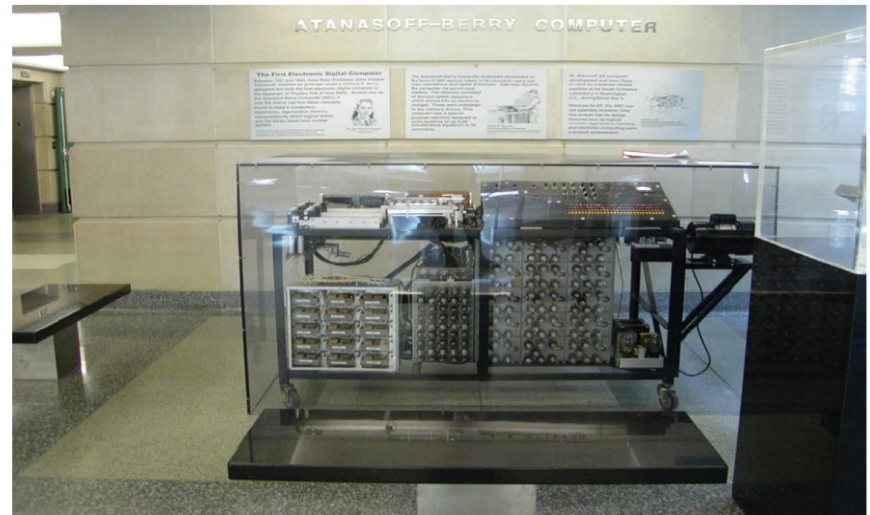


그림 1.15 ABC 컴퓨터

❖ Z4

- 콘라드 추체가 발명 (1945)
- 최초의 범용 전자식 디지털 컴퓨터 제작 공로 인정
- 세계 최초의 상업적인 디지털 컴퓨터
- 내장 프로그램 개념을 사용하지 않음
- 플란칼쿨(Plankalkul)이라는 프로그래밍 언어 사용

❖ 맨체스터 소규모 실험기계

(Manchester Small-Scale Experimental Machine: SSEM 또는 Baby)

- 최초의 현대적인 컴퓨터 (1948)
- 저장 프로그램 개념을 사용한 최초의 컴퓨터

❖ 고급 프로그래밍 언어 (high-level programming language) 발명

- 기계어 명령(대부분 숫자)을 사용하는 대신
- 영어와 유사한 명령어에 의존하는 언어

❖ 트랜지스터

- 잭 킬비와 로버트 노이스가 연구 (1950~1960)
- 기존의 진공관의 사용을 대체 더 작고, 더 빠르고, 더 신뢰성 있는 기술

❖ 집적회로 (integrated circuits)

수천 개의 트랜지스터를 단일 칩에 결합시킨 실리콘 웨어퍼 컴퓨터 크기의 소형화

❖ 반도체 기술 (semiconductor technology)

실리콘을 집적회로로 만드는 기술

❖ 칩 (chip)

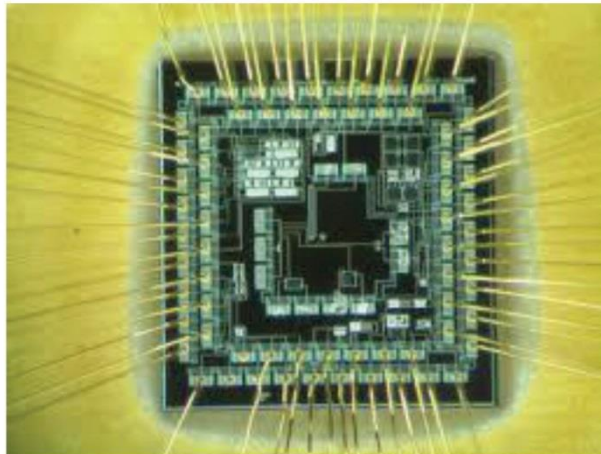


그림 1.16 집적회로

무어의 법칙 (Moore's law)

- 집적회로내의 소자의 수가 18개월마다 대략 두 배씩 증가하도록 제조 능력이 향상된다는 예견 (1965년 논문 발표)
- **지수적 증가** (exponential growth)
 - 2의 지수
- 더 많은 소자들이 동일한 집적회로 공간에 들어갈 때 전류가 흐르는 거리는 더 짧아질 수밖에 없음
 - 회로가 더 빨라짐

동영상 (35s) , <https://www.youtube.com/watch?v=NyW725EUzaE>

1.7 무어의 법칙

18개월 후	개선 요인 = $2^1 = 2$
3년 후	개선 요인 = $2^2 = 2 \times 2 = 4$
4.5년 후	개선 요인 = $2^3 = 2 \times 2 \times 2 = 8$
6년 후	개선 요인 = $2^4 = 2 \times 2 \times 2 \times 2 = 16$
7.5년 후	개선 요인 = $2^5 = 2 \times 2 \times 2 \times 2 \times 2 = 32$
9년 후	개선 요인 = $2^6 = 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 64$
10.5년 후	개선 요인 = $2^7 = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 128$
12년 후	개선 요인 = $2^8 = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 256$
13.5년 후	개선 요인 = $2^9 = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 512$
15년 후	개선 요인 = $2^{10} = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 1,024$

계산적 사고에 대한 정의 (정리)

❖ 연구자의 관심분야에 따라 계산적 사고의 정의는 약간씩 다른 측면을 보이지만 공통적인 의견을 요약

- ① 컴퓨터 과학에 기초한 문제해결의 방법이다.
- ② 문제해결의 논리적 구체적, 계산형태의 명확한 기호적 표현 방법이다.
- ③ 문제해결의 효율적 실행을 위한 절차적 사고의 과정이며 인간의 지적 능력을 확장하는 추론의 과정이다.
- ④ 문제해결을 위해 하드웨어/소프트웨어를 인지적 도구로 사용한다.
- ⑤ 기술과 사고 과정의 한정된 집합이 아니라, 기술과 학습의 역동적 성격을 반영하는 개방적이고 성장하는 개념이다
- ⑥ 핵심적 개념은 명확한 절차의 반복 적용, 탐색, 패턴 매칭, 반복적 정교화, 무작위 기법, 모델링, 시뮬레이션, 시각화 등이다.
- ⑦ 교육의 대상은 공학자, 과학자를 중심으로 전 분야의 대학생, 대학원생, 그리고 K-12과정의 학생들이다.

- ❖ 계산적 사고의 핵심은 프로그래밍이 아닌 개념화에 있다.
 - 여러 단계의 추상화를 통해 사고하는 것
- ❖ 계산적 사고는 단순 반복적인 기술이 아닌 모든 사람이 갖추어야 할 핵심 역량이다.
 - 인간처럼 사고하는 컴퓨터
- ❖ 계산적 사고는 컴퓨터가 아닌 인간의 사고 방식이다.
 - 컴퓨터 기기에 인간의 영리함을 불어넣어 상상도 못한 문제를 해결
- ❖ 계산적 사고는 수학적 사고와 공학적 사고를 보완하고 결합한다.
- ❖ 계산적 사고는 인공물이 아닌 아이디어이다.
 - 문제를 해결하기 위해, 이상 생활을 꾸려나가기 위해, 다른 이들과 소통하기 위해 발전된 컴퓨팅 적 개념 또한 우리의 삶을 구석구석에 막대한 영향을 끼치고 있다.
- ❖ 계산적 사고는 모두를 위한 것이다.
 - 계산적 사고가 인간 활동에 필수가 되어 자연스러운 삶의 일부가 될 것이다.

❖ 재귀적 사고

- 문제해결의 방법을 찾은 뒤, 동일 한 방법이 완전히 성공할 때까지 계속 반복 적
용할 수 있는 사고

❖ 개념화

- 단순히 소스 코드를 프로그래밍하는 것이 아니라 분석, 설계, 코딩 등의 여러 단계
에서 추상화 시각으로 접근할 수 있는 사고

❖ 분해

- 어려운 문제를 작게 쪼개고 분할하여 해결할 수 있는 사고

❖ 추상화

- 복잡한 문제의 공통적인 부분을 인식하여 통합한 뒤 파악할 수 있는 사고

❖ 병렬 처리

- 대역폭이 넓게 통합적인 시각에서 문제를 병행적으로 파악하고 처리할 수 있는
사고

1.8 참고문헌

- 생각하는 프로그래밍, 윤성준,조상민 역, 인사이드, 2014
- 벤츠 타는 프로그래머, 정금호 저, 제이펍, 2013
- 컴퓨터과학이 여는 세계, 이광근 저, 인사이드, 2015
- 소프트웨어 전쟁, 백일승 저, 더하기북스, 2015
- 김대수, "소프트웨어와 컴퓨팅 사고", 생능출판, 2017
- 박두순, 순천향대학교 컴퓨팅적 사고 강의노트, 2017

Q & A

➤ 발표방식: A/B그룹으로 나누어 랜덤 발표

- A그룹 (1~9조), B그룹(10~마지막조)
- A그룹: 문제 1번, B그룹: 문제 2번 에 대하여 PPT 자료 활용 15분 발표준비 (발표 10분 내외, 질의응답 5분)
- 조별 질문 1개 이상씩 준비 필수
- 좌석배치: 앞 1~2라인 각 조별 조장 착석 (1교시 종료 후, 좌석 배치)
- 모든 조는 발표자료를 수업전 일요일 저녁 7시까지 e클래스에 업로드 완료해야함 (시간 및 파일명 엄수!!)
- 파일명 : 날짜-문제번호-조번호
예) 20170910-01-01 // 1조가 문제1번을 발표자료를 준비함

➤ 문제 1

- 컴퓨터 소프트웨어는 의료, 군대, 자동차, 스마트폰 등 다양한 분야에서 사용되어 우리 사회에 지대한 영향을 미치고 있다. 특정 분야를 한가지 선택한 후, 이러한 소프트웨어의 영향을 발표하라.

➤ 문제 2

- 최근 IT 기술의 급속한 발전에 따라 컴퓨터가 빠른 속도로 진화하고 있다. 일반적으로 지금까지의 컴퓨터는 1~4세대의 4 가지 세대로 구분되고, 미래의 컴퓨터는 5세대 컴퓨터로 불리고 있다. 1/2/3/4/5세대 컴퓨터의 특징을 각각 조사하여 발표하라.