



# 자료구조



강 의 노트

## 자료구조와 알고리즘의 이해

---

## 학습내용

- ❖ 프로그램의 기본
- ❖ 자료구조
- ❖ 알고리즘

## 학습목표

- ❖ 프로그램을 할 때 필요한 자료구조의 기본에 대해 설명할 수 있다.
- ❖ 프로그램에서 쓰이는 자료구조의 개념을 설명할 수 있다.
- ❖ 프로그램에서 사용되는 알고리즘의 개념을 예를 들어 설명할 수 있다.



## 프로그램의 기본

### 1. 프로그램의 기본

#### 1) 프로그램 능력이란?

- 요구되는 작업을 수행하기 위해 컴퓨터 S/W를 다루는 능력



#### 2) 프로그램 능력을 갖추기 위해 선행하여 갖추어야 할 내용

- 컴퓨팅, 객체, 타입, 입출력 장치의 역사
- ++ 연산자, 반복문, 함수, 인자 전달 방법
- 배열과 문자열
- 포인터와 연결리스트
- Sorting 알고리즘
- 객체 지향 설계 및 추상 클래스

#### 3) 프로그램과 자료구조

“프로그램 개발 시 어떤 자료구조를 선택해야 할까?”





프로그램의 기본

2. 자료 저장 기법

1) 배열(Array)



- 동일한 형태의 자료
- 메모리에 연속적으로 정적 할당
- 단순하지만 처리 효율은 떨어짐

2) 연결리스트(Linked List)



- 재귀적 자료 구조
- 메모리에 불연속적으로 동적 할당
- 복잡하지만 처리 효율이 우수




자료구조란?

1. 자료구조란?

1) 자료구조(Data Structure)의 정의

- 자료를 효율적으로 컴퓨터에 저장하는 방법

효율적인 자료구조?



- 궁극의 자료구조는 존재하지 않음
- 요구사항에 의존적으로 선택

2. 프로그램

1) 프로그램 = 자료구조 + 알고리즘

예) 최댓값 탐색 프로그램

= 배열 + 순차탐색

score[ ]

80	70	90	...	30
----	----	----	-----	----

```
tmp ← score[0];
for i ← 1 to n do
    if score[i]>tmp
        then tmp ← score[i];
```



알고리즘

1. 알고리즘의 개요

1) 정의



알고리즘(Algorithm)

- 컴퓨터로 문제를 풀기 위한 단계적인 절차

2) 알고리즘의 조건

- 입력 : 0개 이상의 입력이 존재해야 한다.
- 출력 : 1개 이상의 출력이 존재해야 한다.
- 명백성 : 각 명령어의 의미는 모호하지 않고, 명확해야 한다.
- 유한성 : 한정된 수의 단계 후에는 반드시 종료되어야 한다.
- 유효성 : 각 명령어들은 실행 가능한 연산이어야 한다.

2. 알고리즘의 기술 방법

1) 자연어

- 예) 영어, 한국어
  - 읽기 쉬움
  - 단어들을 정확하게 정의하지 않으면 의미전달이 모호

2) 흐름도

- 흐름도    예) Flow Chart
  - 읽기 쉬움
  - 단어들을 정확하게 정의하지 않으면 의미전달이 모호



## 알고리즘

### 2. 알고리즘의 기술 방법

#### 3) 유사코드

- 유사코드 예) Pseudo - Code
  - 알고리즘 기술에 가장 많이 사용
  - 프로그램을 구현할 때의 여러 가지 문제들을 감출 수 있음  
(알고리즘의 핵심적인 내용에만 집중)

#### 4) 프로그래밍 언어

- 프로그래밍 언어 예) C, C++
  - 알고리즘의 가장 정확한 기술이 가능
  - 실제 구현 시 많은 구체적인 사항들이 알고리즘의 핵심적인 내용에 대한 이해를 방해



알고리즘

3. 알고리즘의 예시

[예1] 배열이나 연결리스트에서 k번째 원소를 가져오는 경우

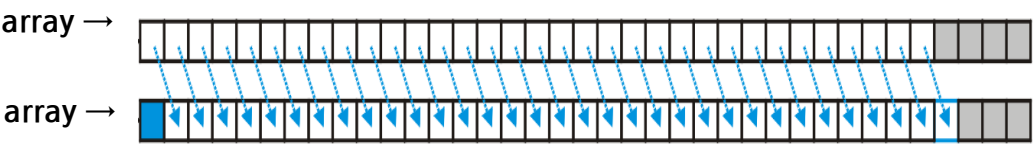
- 배열에서는 첨자를 이용하여 `array[k]`처럼 한 번에 가능(1회)
- 연결리스트에서는 `k-1` 노드들을 다 거쳐가야 함(`k-1`회)

[예2] 소팅된 배열이나 연결리스트에서 어떤 원소를 찾는 경우

- 소팅된 배열에서는 이진 탐색 기법을 사용(매우 빠름)
- 다른 일반적인 방법으로는 찾는 원소보다 작은 원소를 다 살펴봐야 함(느림)

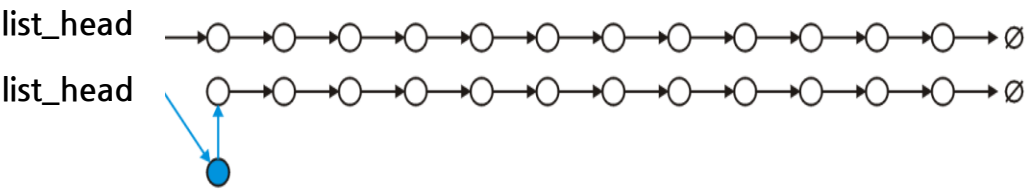
[예3] 배열에서 새 원소를 삽입하는 경우

- 배열에서는 삽입 후에 자리가 바뀐 모든 원소를 다시 복사(매우 느림)



[예4] 연결리스트에서 새 원소를 삽입하는 경우

- 연결고리를 재 배치(크기에 관계 없이 동일한 시간 소요, 매우 빠름)





## 핵심정리

### 프로그램의 기본

- 요구되는 작업에 대해 컴퓨터 S/W를 다루는 능력
- C++ 프로그래밍의 기본적 지식 필요

### 자료구조

- 프로그램에서 자료들을 정리하기 위한 여러 가지 유용한 구조

### 알고리즘

- 컴퓨터로 문제를 풀기 위한 단계적인 절차
- 알고리즘 기술 방법: 자연어, 흐름도, 유사코드, 프로그래밍 언어(C 등)