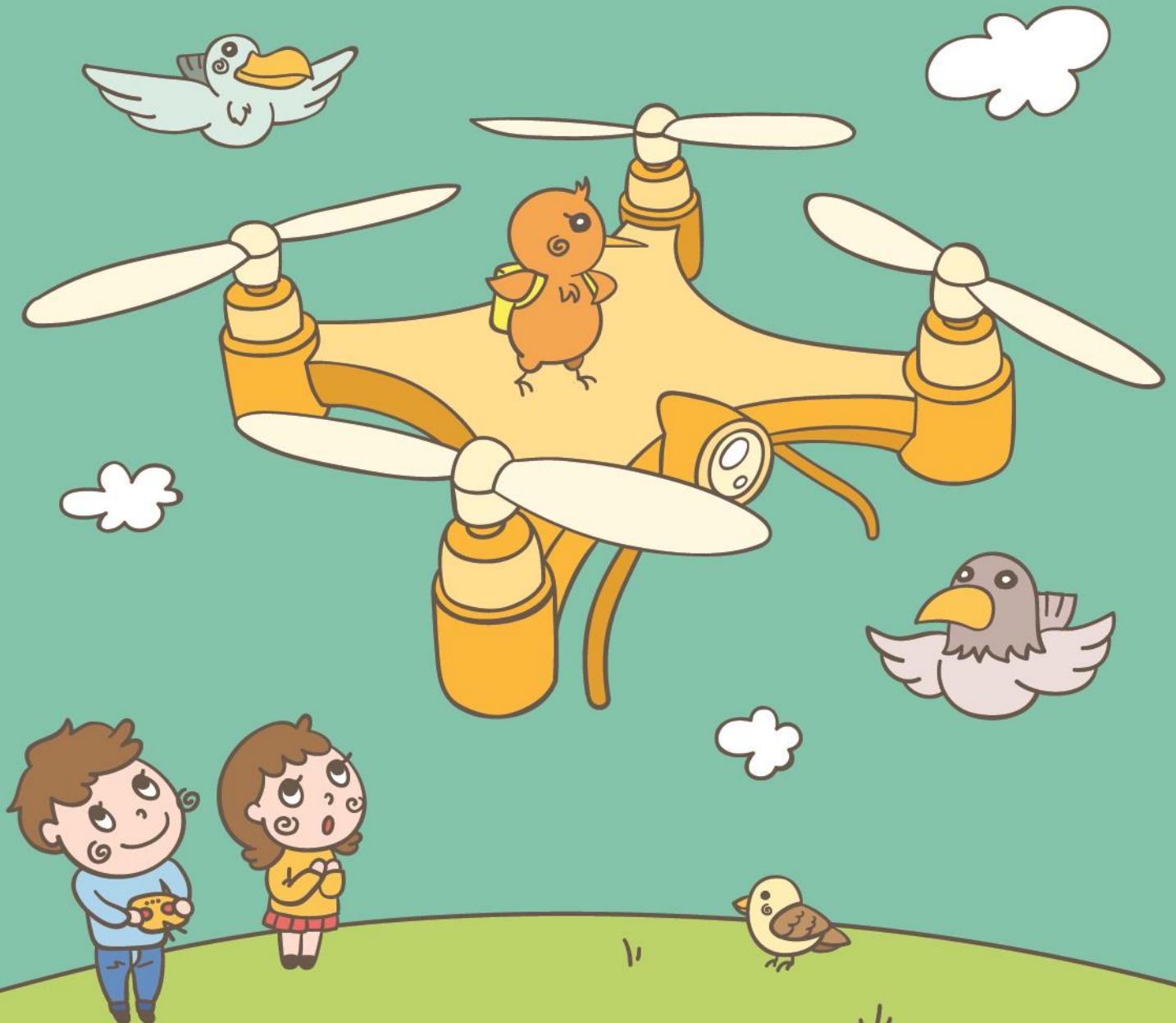




드론으로 배우는
프로그래밍 교실

Ch1. 소프트웨어의 이해 - 3



01 알고리즘과 친해지기	01
알고리즘이란?	02
알고리즘 퀴즈	04
알고리즘 세부 내용	05
02 알고리즘 종류 살펴보기	06
알파고와 알고리즘	07
정렬 알고리즘	08
압축 알고리즘	10
03 알고리즘 만들어 보기	12
내가 만드는 알고리즘	13



드론으로 배우는
프로그래밍 교실

초판발행 2016년 9월 23일
지은이 최정애 | 펴낸이 최정애
펴낸곳 WHIT | 주소 안산시 한양대로55 창업보육센터 B01
전화 010-5125-2139

Published by WHIT. Printed in Korea
Copyright © 2016 최정애 & WHIT

이 책의 저작권은 최정애와 WHIT에 있습니다.
저작권법에 의해 보호를 받는 저작물이므로 무단 복제 및 무단 전재를 금합니다.

01 알고리즘과 친해지기



알고리즘은 프로그래밍을 하기 위한 기본입니다.

단순히 코딩을 하는게 아닌 문제를 해결해 나가는 과정을 직접적으로 배우고 사용할 수 있게 해주는게 알고리즘입니다.

문제를 논리적으로 풀어 나가는 과정을 통해 컴퓨팅 사고력을 기르며, 다양한 분야에 사용되는 알고리즘을 알아가 봅시다.

알고리즘
정의

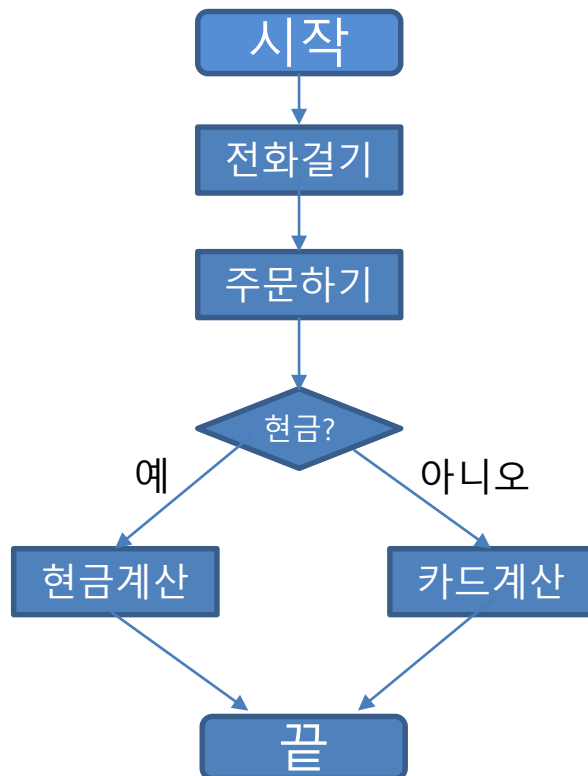
알고리즘

알고리즘이란 어떤 문제를 해결하기 위한 여러 동작들의 모임입니다. 쉽게 설명하면, 문제를 해결 해 나가는 과정을 나열해 놓은 것이라고 할 수 있습니다.

예를 들어, 피자를 주문하는 과정은

- ① 전화를 건다.
- ② 주문을 한다.
- ③ 카드인지 현금인지 대답한다.
- ④ 카드 또는 현금으로 계산한다.
- ⑤ 피자를 받는다.

와 같은 동작들로 구성됩니다.




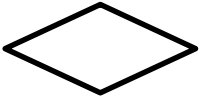





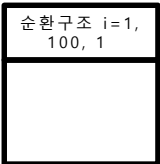
알고리즘
순서도

<그림1-1> 순서도 예시

순서도

순서도는 미리 정의된 기호와 그들을 서로 연결하는 선을 사용하여 프로그래밍의 여러 문제를 해결하는 데 필요한 논리적인 흐름을 그림으로 표현한 것이다. 출처: 네이버 백과사전

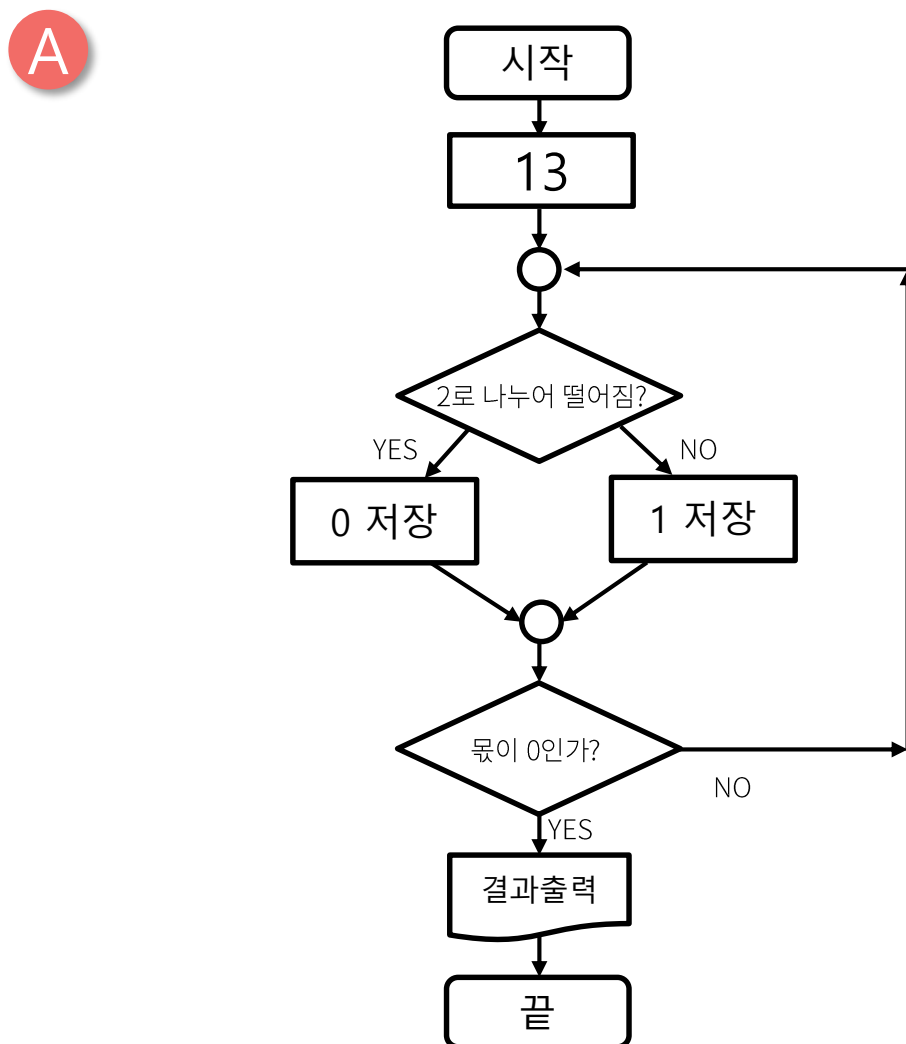
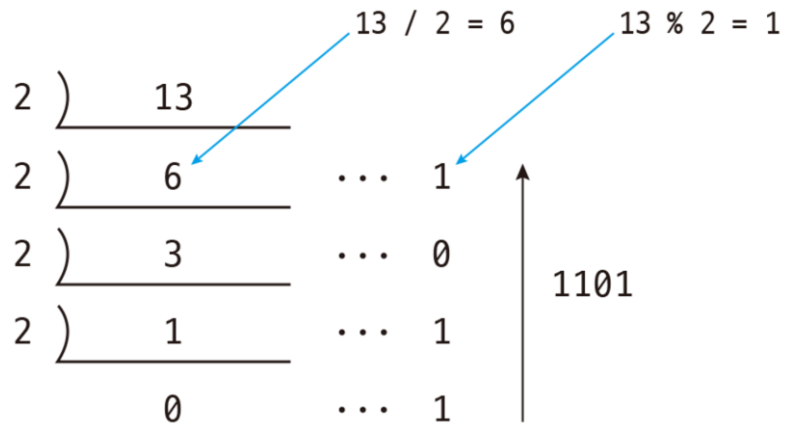
순서도 구성

기호	이름	의미
	단자	순서도의 시작과 끝
	준비	변수 선언 및 초기화
	처리	값을 계산하거나 대입
	판단	참과 거짓을 판단
	수동입력	키보드로 입력
	입출력	데이터 입출력
	문서	결과를 프린터로 출력
	흐름선	처리 기호와 흐름을 연결
	연결자	다른 곳으로의 연결
	순환구조	반복문

<그림1-2> 순서도 구성

알고리즘 맛보기

Q 13을 2진수로 바꾸는 과정을 알고리즘 순서도로 그리시오.



알고리즘
조건

알고리즘은 다음 조건을 만족해야 한다.

- ① 입력 : 외부에서 제공되는 자료가 있을 수 있다.
- ② 출력 : 적어도 한 가지 결과가 생긴다.
- ③ 명백성 : 각 명령들은 명백해야 한다.
- ④ 유한성 : 알고리즘의 명령대로 수행하면 한정된 단계를 처리한 후에 종료된다.
- ⑤ 효과성 : 모든 명령들은 명백하고 실행 가능한 것이어야 한다.

알고리즘
분석기준

- 정확성 : 알고리즘에서 수행 될 입력과 출력에 대해 정확히 이해
- 작업량 : 문제를 해결하기 위해 필요한 양을 최소화
- 기억장소사용량 : 데이터를 저장하기 위한 공간의 필요
- 단순성 : 알고리즘을 간단하게 디자인 할 필요
- 최적성 : 같은 문제를 해결하는데 더 적은 연산을 수행하는 것

알고리즘
복잡도

- 시간 복잡도 : 알고리즘을 구성한 명령어가 실행 된 횟수
- 공간 복잡도 : 알고리즘을 구성하는데 사용된 메모리

빅오
표기법

시간 복잡도에서 불필요한 정보를 제거함으로 알고리즘 분석을 쉽게 할 용도로 시간복잡도를 나타내는 방법을 빅오 표기법이라고 합니다.

Ex) $O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$, $O(n^3)$, ...

02 알고리즘 종류 살펴보기



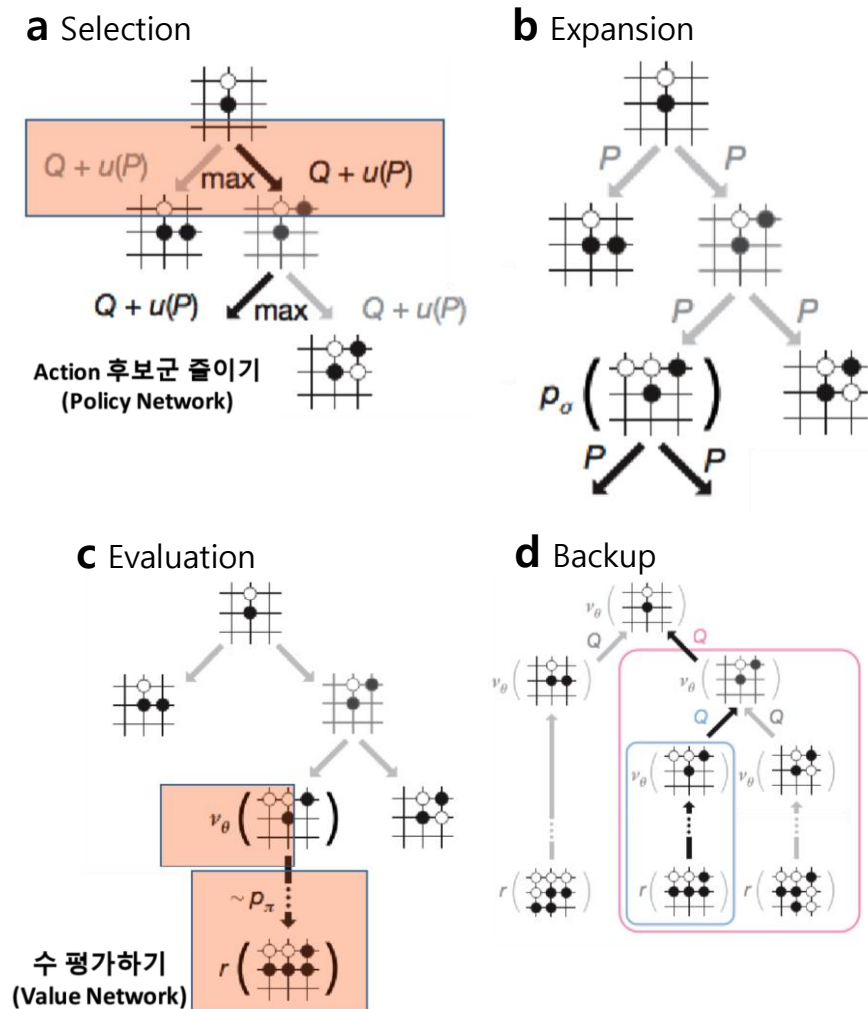
알고리즘은 다양한 분야에서 사용됩니다.

알파고에 쓰인 알고리즘은 머신러닝 알고리즘으로 기계 학습에 대한 문제 해결 방법입니다.

폴더에 이름 순, 연도 순으로 정렬을 할 때에는 정렬 알고리즘이 사용되고, 압축을 할 때 대표적으로 사용되는 알고리즘이 허프만 알고리즘입니다.

다양한 알고리즘 예제를 통해 알고리즘에 한 걸음 더 다가가 봅시다.

알파고 알고리즘



<그림2-2> 알파고에 쓰인 알고리즘

몬테카를로 트리 탐색

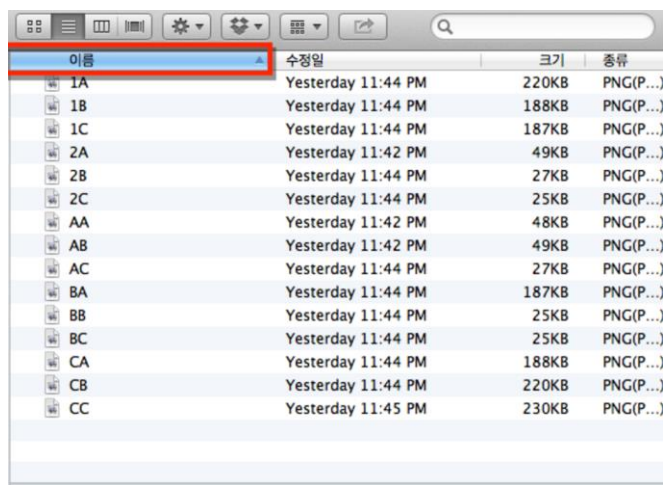
알파고의 알고리즘은 몬테카를로 트리 탐색입니다.

몬테카를로 트리탐색은 최소-최대(Minimax) 알고리즘의 성능을 개선한 것으로 모든 경로를 탐색하기 불가능할 때 효율적입니다.

절차는 선택(Selection), 확장(Expansion), 시뮬레이션(Simulation), 역전파(Backpropagation)란 과정을 거칩니다. 간단하게 말하면 현상태에서 특정 경로로 수읽기를 선택한 뒤, 일정 수 이상 진행되면 한 단계 다음 지점을 예측해 확장합니다.

데이터의
정렬은?

도서관에 가보면 책들이 가나다 순으로 정렬되어 있는 걸 확인할 수 있습니다. 이름 순 이외에 발행연도 순, 크기 순, 저자 순 등 여러 방법으로 책들을 정렬할 수 있습니다. 그렇다면 컴퓨터에 들어 있는 데이터들은 어떤식으로 정렬을 하게 될까요?



이름	수정일	크기	종류
1A	Yesterday 11:44 PM	220KB	PNG(P...)
1B	Yesterday 11:44 PM	188KB	PNG(P...)
1C	Yesterday 11:44 PM	187KB	PNG(P...)
2A	Yesterday 11:42 PM	49KB	PNG(P...)
2B	Yesterday 11:44 PM	27KB	PNG(P...)
2C	Yesterday 11:44 PM	25KB	PNG(P...)
AA	Yesterday 11:42 PM	48KB	PNG(P...)
AB	Yesterday 11:42 PM	49KB	PNG(P...)
AC	Yesterday 11:44 PM	27KB	PNG(P...)
BA	Yesterday 11:44 PM	187KB	PNG(P...)
BB	Yesterday 11:44 PM	25KB	PNG(P...)
BC	Yesterday 11:44 PM	25KB	PNG(P...)
CA	Yesterday 11:44 PM	188KB	PNG(P...)
CB	Yesterday 11:44 PM	220KB	PNG(P...)
CC	Yesterday 11:45 PM	230KB	PNG(P...)

<그림2-3> 이름 순으로 정렬

정렬
알고리즘

데이터를 정렬할 때 사용하는 알고리즘을 정렬 알고리즘이라고 합니다. 대표적인 정렬 알고리즘은 아래의 다섯가지 입니다.

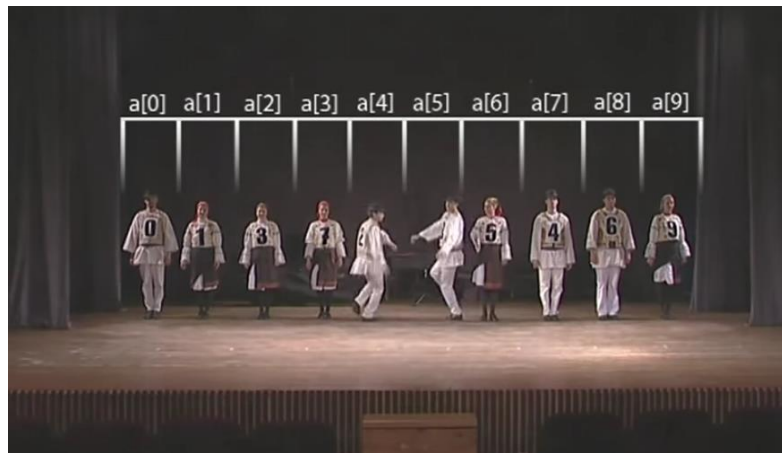
- 버블 정렬
- 선택 정렬
- 합병 정렬
- 삽입 정렬
- 퀵 정렬

선택 정렬

선택 정렬은 이름처럼 현재 위치에 들어갈 값을 찾아 정렬하는 것입니다. 현재 위치에 저장될 값이 최솟값인지 최댓값인지에 따라 오름차순 정렬인지 내림차순 정렬인지가 정해집니다.

버블 정렬

버블 정렬은 인접한 두 데이터를 비교하여 정렬하게 된다. 오름차순으로 정렬할 경우 두 데이터를 비교하여 우측이 크면 움직이지 않고, 우측이 작으면 두 데이터를 바꾸어 놓는다. 이렇게 진행하게 되면 큰 데이터는 가장 끝으로 가게 된다.



<그림2-4> 버블 정렬

출처 : <https://www.youtube.com/watch?v=ebi54DXYG8>

합병 정렬

합병 정렬은 데이터들을 둘 또는 하나가 될 때까지 계속 나눈 후 나뉜 원소들을 비교하여 합쳐 나가면서 정렬을 진행 한다. 병합 정렬이라고도 불린다.

삽입 정렬

삽입 정렬은 데이터 중 한 원소를 정한 후(보통 가장 좌측) 그 원소들을 우측에 있는 원소들과 비교해가며 자신이 작을 경우 움직이지 않고, 자신이 클 경우 그 원소와 자리를 바꾸어 가면서 정렬을 진행 한다.

퀵 정렬

퀵 정렬은 데이터 중 한 원소를 정한 후 그 원소의 좌측에 있는 값이 자신보다 크면서, 동시에 그 원소의 우측에 있는 값이 자신보다 작으면 주 원소를 교환하는 식으로 정렬을 진행 한다.

허프만
알고리즘

허프만 알고리즘은 압축 알고리즘 중 가장 대표적인 알고리즘입니다.
허프만 알고리즘에 대해 알아보기 전에 압축이란 무엇인지부터 짚고 넘어가겠습니다.

압축이란?

파일 압축 프로그램을 사용하여 프로그램 파일이나 데이터 파일의 크기를 축소함으로써 파일을 저장하는 디스크의 기억 공간을 절감하고 전송하는 데 필요한 시간을 단축하기 위한 작업 출처 : Naver 지식백과

허프만
알고리즘
원리

허프만 알고리즘은 마치 줄임말과 암호를 섞은 듯한 모양입니다. 예를 들어 '비번'은 '비밀번호'를 줄인 단어로 4글자를 2글자로 줄여서 표현한 것입니다.
이와 마찬가지로 자주 나오는 단어를 짧은 비트로 치환하여 표현하는 것이 허프만 알고리즘의 핵심입니다.

허프만
알고리즘
예시

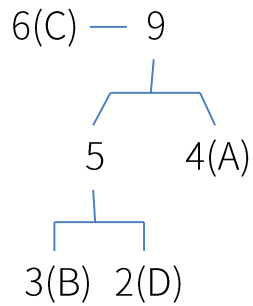
DBACDCBCCACAACB 라는 문자열을 압축 해 봅시다.
우선, 각 문자의 빈도 수를 파악해야 합니다.

A : 4, B : 3, C : 6, D : 2 이후 빈도수대로 늘어놓습니다.

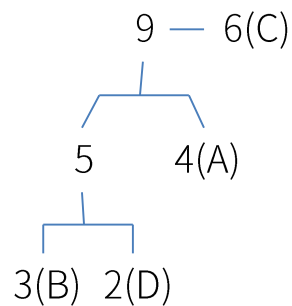
6(C), 4(A), 3(B), 2(D) 이후 하위 두 개를 묶습니다.

6(C) — 4(A) — 5
 └─┬─┘
 3(B) 2(D) 이후 빈도수대로 늘어놓습니다.

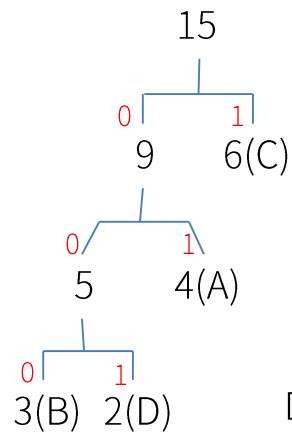
6(C) — 5 — 4(A) 이후 하위 두 개를 묶습니다.
 └─┬─┘
 3(B) 2(D)



이후 빈도수대로 늘어놓습니다.



이후 하위 두 개를 묶습니다.



이후 오른쪽에 1을, 왼쪽에는 0을 붙여 넣습니다.

이후 15부터 시작하여 각 알파벳에 비트를 붙입니다.

A : 01, B : 000, C : 1, D : 001

DBACDCBCCACAACB라는 문자열이
001 000 01 1 001 1 000 1 1 01 1 01 01 1 000
으로 치환되었습니다.

기존의 문자열은 한 문자 당 1byte를 차지하여 총 15byte 즉 120bit였으나 압축 후 29 bit로 약 4byte가 안되게 줄어들었습니다.
헤더 부분을 제외하고 약 24퍼센트로 줄어들어 큰 압축율을 보입니다.

03 알고리즘 만들어 보기



이번 장에서는 앞서 배운 알고리즘을 통해 자신이 직접 알고리즘을 만들어 볼 것입니다.

일상 생활에서 일어나는 문제를 논리적이고 순차적으로 작성하는 활동을 통해 알고리즘이 멀리 있지 않고 우리 생활과 밀접한 연관이 있다는 것을 느껴 보세요. 알고리즘을 직접 만들다 보면 어느새 논리적으로 생각하고 있는 자신을 발견할 수 있을 겁니다.

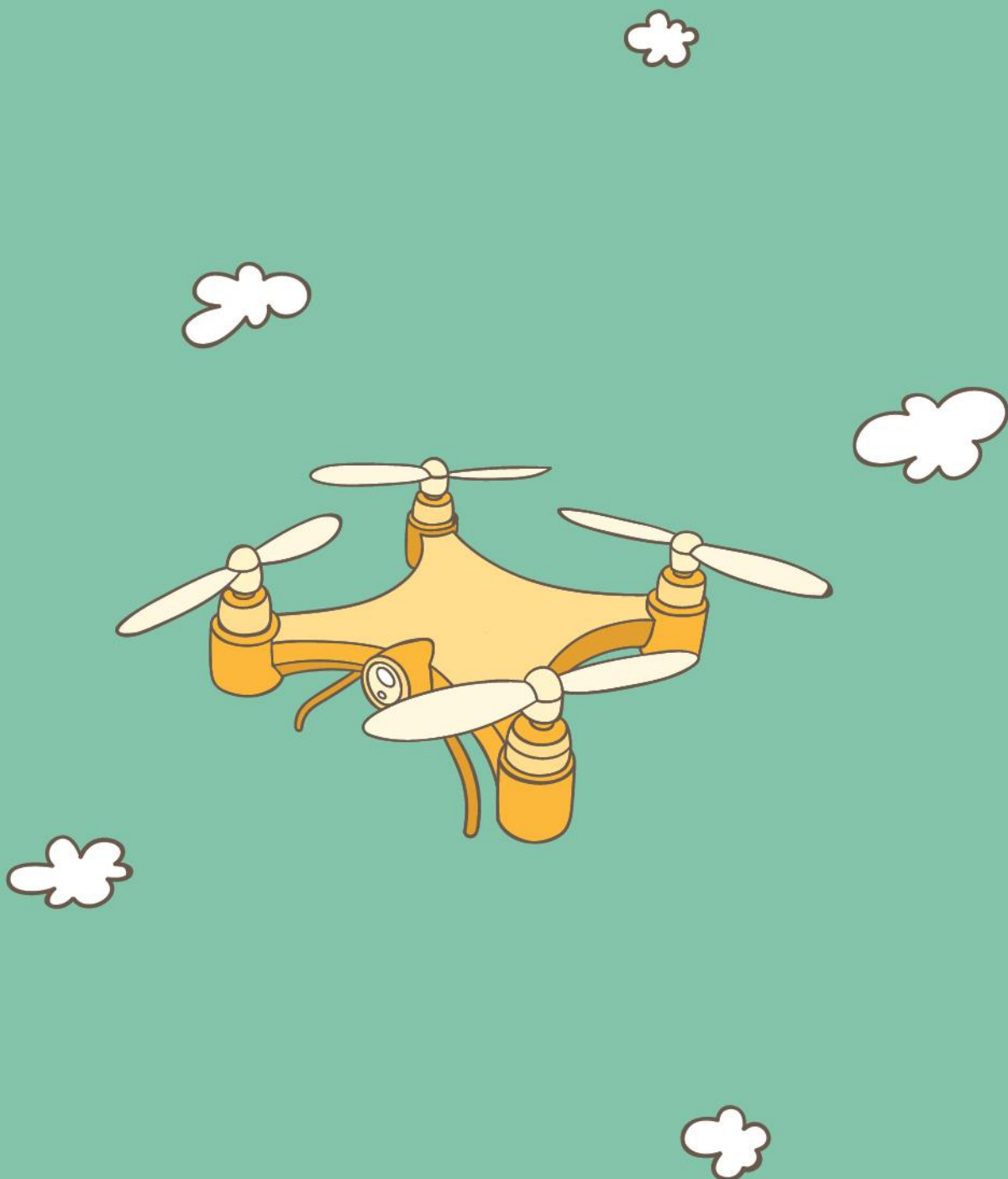
내가 만드는 알고리즘

알고리즘
순서도
만들기

자신만의 알고리즘 순서도를 만들어 보세요. 일상 생활에서 일어나는 모든 일들이 알고리즘이 될 수 있습니다.
Ex) 도둑과 마주쳤다, 나는 배가 고프다.

시작

끝



WHIT