

01. 자료구조와 알고리즘

1. 1 자료구조와 알고리즘

$$\text{프로그래밍} = \text{자료구조} + \text{알고리즘}$$

자료 알고리즘

알고리즘 (Algorithm) : 컴퓨터를 통제하는 풍기 규칙을 단계적인 절차

문제를 컴퓨터가 처리할 수 있도록 통제를 자료구조와 알고리즘으로, 정확하게 작동하는 프로그램을 만들 수 있는 언어를 기술하는 것

복잡한 일을 솔직하게 명령어들의 절차,

입력 : 0이 아님의 입력이 들어온다면 1로 바꾸기

알고리즘 기술 방법

출력 : 1이 아님의 출력이 들어온다면 0로 바꾸기

① 흐름, 의사의 과정

복잡화 : 각 명령어의 처리는 단순화되며 알고 명령어로는 처리된다

② 흐름도 (Flowchart)

구현화 : 프로그램 수식으로 쉽게 주어진 뒤 다시 풀어서 구현화되어야 한다

③ 의사코드 (Pseudo-code)

구현화 : 각 명령어들을 종이에 옮기고, 그는 컴퓨터로 실행 가능하도록 만든다

④ 프로그래밍 언어

1. 2 추상 자료형

추상 자료형 (ADT: abstract data type) : ADT의 이름, 객체 (objects), 함수 (functions)

1. 3 알고리즘의 성능 분석

알고리즘 복잡도 분석 (Complexity analysis) < 시간 복잡도 (Time complexity) : 수행 시간

공간 복잡도 (Space complexity) : 알고리즘이 차지하는 공간

복선 풀기법 O(n) : 가장 큰 차수에 영향을 받는 풀기법 (상수)

$$O(1) > O(\log n) > O(n) > O(n \log n) > O(n^2) > O(n^3) > O(2^n) > O(n!)$$

복잡도로 가능한 시간이 오래 걸림

복선 풀기법 풀기법 $\Sigma (n)$: 가장 작은 차수에 영향을 받는 풀기법 (하나)

복선 풀기법 $\Theta(n)$: 가장 큰 차수와 차수에 영향을 받는 풀기법 (상·하한)

최선의 경우 (best case) : 수행 시간은 가장 빠름, 의미 X

평균의 경우 (average case) : 수행 시간은 평균적인 경우를 의미함

최악의 경우 (worst case) : 수행 시간은 가장 느림, 널리 불편