

Computer Algorithm Basic-2

Analyze to Computer Algorithm

컴퓨터 알고리즘 4단계

1. 문제정의

컴퓨터를 이용해서 어떤 문제를 풀것인가.

2. 알고리즘 설명

요리법처럼 단계적으로 설명하는 것

3. 정확성 증명

4. 성능 분석

컴퓨터알고리즘에서 가장중요.

문제를 효율적으로 해결할것인지가 중요한 포인트 중에 하나.

성능분석에 대해 알아보자!

문제를 해결하는 하나의 논제가 주어졌을 때 해결하는 방법이 많음.

ex) 목적지에 도달하는 방법

인프라, 시간, 비용 등에 따라 여러가지 방법이 나온다.

-성능분석 관점에 알고리즘

1. 문제를 푼다.

2. 문제를 풀 경우 가장 좋은 성능이 뭔지 알아본다.

컴퓨터 알고리즘의 분석

ex)

$34534546 * 2020202020 * 3466342462649$?

장비:: 스마트폰, 노트북, 슈퍼컴퓨터? → 환경, 장비에 따라 성능의 측정이 달라진다

조건이 동일한 특정 기계에서 모든 알고리즘의 수행시간을 측정해야하는데

현실적으로 불가능. 따라서 수행 연산 횟수를 비교하는 방식으로 성능을 분석한다.

수행 연산의 횟수를 비교하는게 왜 효과적일까?

ex)

안드로이드 휴대폰 10초

PC 1초

Server 0.1초

곱셈연산을 s번을 수행하고 그 곱셈을 10번정도 반복

안드로이드 1초

pc 0.1 초

Server 도 0.01초

공통점 수행연산

차이점 곱셈시간

결국 어떤 알고리즘을 이용해 연산을 푼다는 것은 기본 함수를 몇번 수행하는가를 볼 수 있다.

곱셈연산은 머신 기반. 따라서 어떤 기계에 따라 측정하는 값이 다르다

하지만 횟수가 같기 때문에 알고리즘 성능 평가할때 몇번의 연산을 통해 그 문제를 푸는가에 대해 기준을 둔다.

좋은 알고리즘이라고 하면, 곱셈에 대한 시간이 아닌 횟수를 줄이는 것이 목표가된다.

횟수를 생각하면 문제를 정의했을 때

입력과 출력을 정의했을 시

입력 크기가 크면 시간이 많이 커진다.

입력의 갯수를 몇개로 할지에 대해서도 중요하다.

입력의 갯수를 임의의 크기 n 으로 표시한다.

$T(n)$ 으로 표현할 시 수행시간 분석은 가장 기본이 되는 함수가 무엇인가 판별한 다음에

판별한 함수를 수행하는데 기본 성능을 분석하고 입력의 횟수를 항상 n 으로 정해놨다고 생각하면 된다.

기본이 되는 연산에는 뭐가 있을까?

-성능분석 비교대상

산술 연산 = > 사칙연산 (기본적인 핵심이라고 생각하고있다)

데이터 입출력 = > 복사 움직임 저장 열기 엄청 크다.

제어 연산 = > if while register 브랜치가 생기면 나중에 많이들어갈때 중요하다

보통 곱하기 지수승을 기본적인 핵심으로 생각한다.

점근적 표기법

성능분석을 명확하게 표기하기 위한 분석법

알고리즘은 (문제정의-알고리즘설명-정확성증명-성능분석)

성능분석은 기본이 되는 함수가 몇번 시행되는지 상대적으로 비교

이 비교를 빅 오 표기법, 오메가 표기법, 세타 표기법으로 표기한다.

빅 오 표기법

$cg(n)$ 기준이되는함수, $f(n)$ 비교하려고 하는함수

$cg(n)$ 이 위에있다. asygnpthon upper bound 상한선

n 값에 따라 다양하게 변화하지만

n 이 커지더라도 $g(n)$ 값보다 커지지않는다

ex) $3n+1 = O(n)$ 제곱

$3n+1$ 이 $f(n)$ $O(n)$ 제곱 $g(n)$

왜냐면 제곱만큼 커질수 없으니까!

증명하는 방법은 n 제곱으로 둘을 나눈다.

그럼 1과 4사이 위에면 위의 식은 항상 성립한다.

오메가 표기법

$f(n)$ $cg(n)$ 에 대해 $f(n)$ 이 위에 있다. $\Rightarrow f(n)$ 이 느리다.

n 이 어떻게 변한다 하더라도 내려갈수 없는 범위

하한선. 아무리 빨라도 $cg(n)$ 보다는 느리다는 뜻

$f(n)$ 이 제곱 $cg(n)$ 은 n 이었을때

이차식으로 쑥 증가하지만 $cg(n)$ 1차식이기 때문에

2전일때만족한다.

세타표기

점근적 상한 및 하한

빅오랑 오메가 사이 . 아무리 빨라도 $c1$ 보단 느리고 $c2$ 보단 빠름

이 사이에 존재하는 모든 함수 를 세타로 표현할 수 있다.