
시간복잡도

코딩테스트 광탈방지 A to Z : JavaScript - 이선희 @kciter

JS

우리는 프로그램의 성능을 정확히 알 수 있는가?

고려할 것

입력 크기

하드웨어 성능

운영체제 성능

컴파일러 최적화

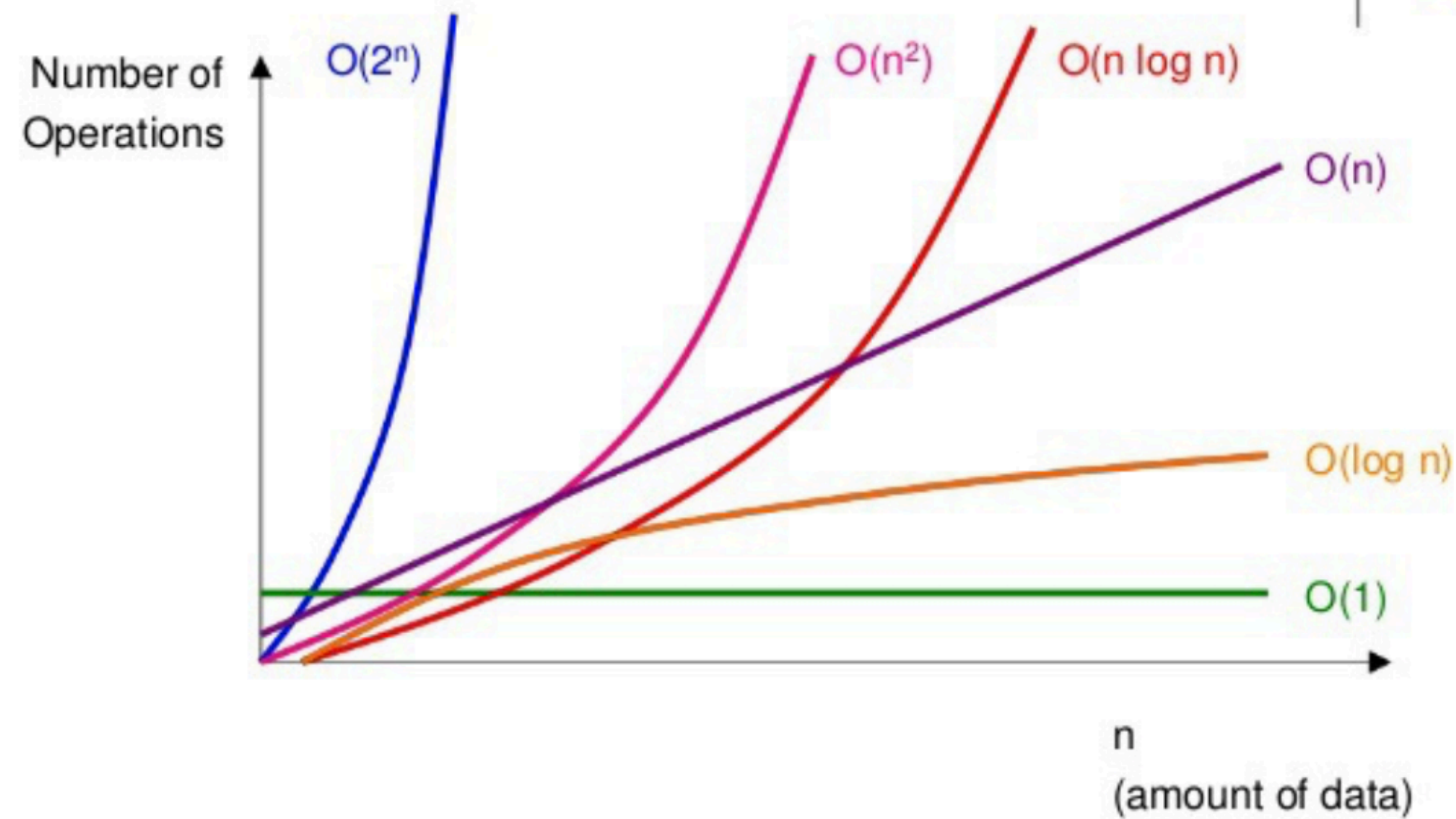
비동기 로직

•
•
•

프로그램의 성능을 정확히 파악하는 것은 불가능...

빅오표기법 Big-O notation

Comparing Big O Functions



(C) 2010 Thomas J Cortina, Carnegie Mellon University

$$O(1) < O(\log n) < O(n) < O(n \log n) < O(n^2) < O(2^n) < O(n!)$$

$O(n)$

```
for (let i = 0; i < n; i += 1) {  
  // ...  
}
```

 $O(\log n)$

```
for (let i = 1; i <= n; i *= 2) {  
  // ...  
}
```

 $O(n \log n)$

```
for (let i = 0; i < n; i += 1) {  
  for (let j = 1; j <= n; j *= 2) {  
    // ...  
  }  
}
```

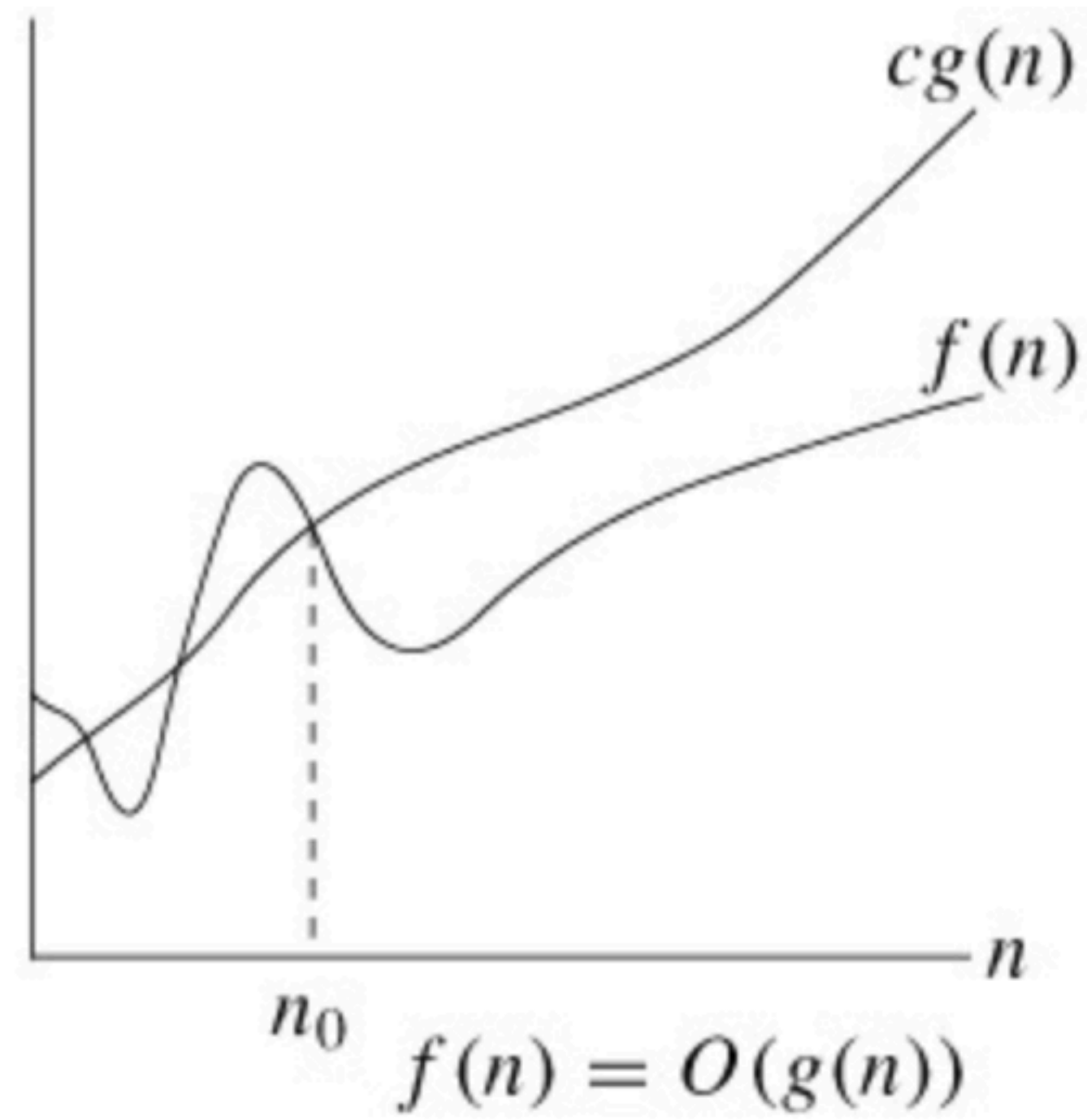
 $O(n^2)$

```
for (let i = 0; i < n; i += 1) {  
  for (let j = 0; j < n; j += 1) {  
    // ...  
  }  
}
```

$$O(n^2 + 126), O(3n - 30), O(3 \log n)$$

is

$$O(n^2), O(n), O(\log n)$$



계수 법칙

상수 k 가 0보다 클 때 $f(n) = O(g(n))$ 이면 $kf(n) = O(g(n))$ 이다.
 n 이 무한에 가까울 수록 k 의 크기는 의미가 없기 때문이다.

```
// 두 루프는 같은  $O(n)$ 으로 표기된다.  
for (let i = 0; i < n; i += 1) {  
  // ...  
}  
  
for (let i = 0; i < n * 5; i += 1) {  
  // ...  
}
```

합의 법칙

$f(n) = O(h(n))$ 이고 $g(n) = O(p(n))$ 이면 $f(n) + g(n) = O(h(n)) + O(p(n))$ 이다.
빅오는 더해질 수 있다.

```
// 두 루프를 합쳐  $O(n + m)$ 으로 표기할 수 있다.  
// 계수 법칙에 의해 5는 사라진다.  
for (let i = 0; i < n; i += 1) {  
  | // ...  
}  
  
for (let i = 0; i < m * 5; i += 1) {  
  | // ...  
}
```

곱의 법칙

$f(n) = O(h(n))$ 이고 $g(n) = O(p(n))$ 이면 $f(n) * g(n) = O(h(n)) * O(p(n))$ 이다.
빅오는 곱해질 수 있다.

```
// 두 루프를 곱해  $O(n^2)$ 으로 표기할 수 있다.  
// 계수 법칙에 의해 5는 사라진다.  
for (let i = 0; i < n; i += 1) {  
  for (let j = 0; j < n * 5; j += 1) {  
    // ...  
  }  
}
```

다항 법칙

$f(n)$ 이 k 차 다항식이면 $f(n)$ 은 $O(n^k)$ 이다.

```
// 다음 루프는  $O(n^3)$ 으로 표기할 수 있다.  
for (let i = 0; i < n * n * n; i += 1) {  
  | // ...  
}
```

2가지만 기억하세요

1 상수항은 무시

```
// 계수 법칙에 의해 계수는 무시된다.  
// 그리하여  $O(n + m)$ 으로 표기된다.  
for (let i = 0; i < n * 6; i += 1) {  
  // ...  
}  
  
for (let i = 0; i < m * 3; i += 1) {  
  // ...  
}
```

2 가장 큰 항 외엔 무시

```
//  $O(n^2 + n)$ 이지만 작은 항은 무시하여  
//  $O(n^2)$ 으로만 표기해도 된다.  
for (let i = 0; i < n; i += 1) {  
  // ...  
}  
  
for (let i = 0; i < n; i += 1) {  
  for (let j = 0; j < n; j += 1) {  
    // ...  
  }  
}
```

성능 측정 방법

Date 객체를 이용

```
const start = new Date().getTime();  
  
// ...  
  
const end = new Date().getTime();  
console.log(end - start);
```

```
console.log("Start");
const start = new Date().getTime();
const N = 1000000000;

let total = 0;
for (let i = 0; i < N; i += 1) {
  total += i;
}

const end = new Date().getTime();
console.log(end - start);
console.log("Finish");
```



Output

```
Start
1114
Finish
```

시간복잡도

코딩테스트 광탈방지 A to Z : JavaScript - 이선희 @kciter

JS