

計算量を学ぼう！



計算量を学ぼう！

ぱうえる（けんた）

2022/12/20 ゼロイチゼミ @nu_zero_one

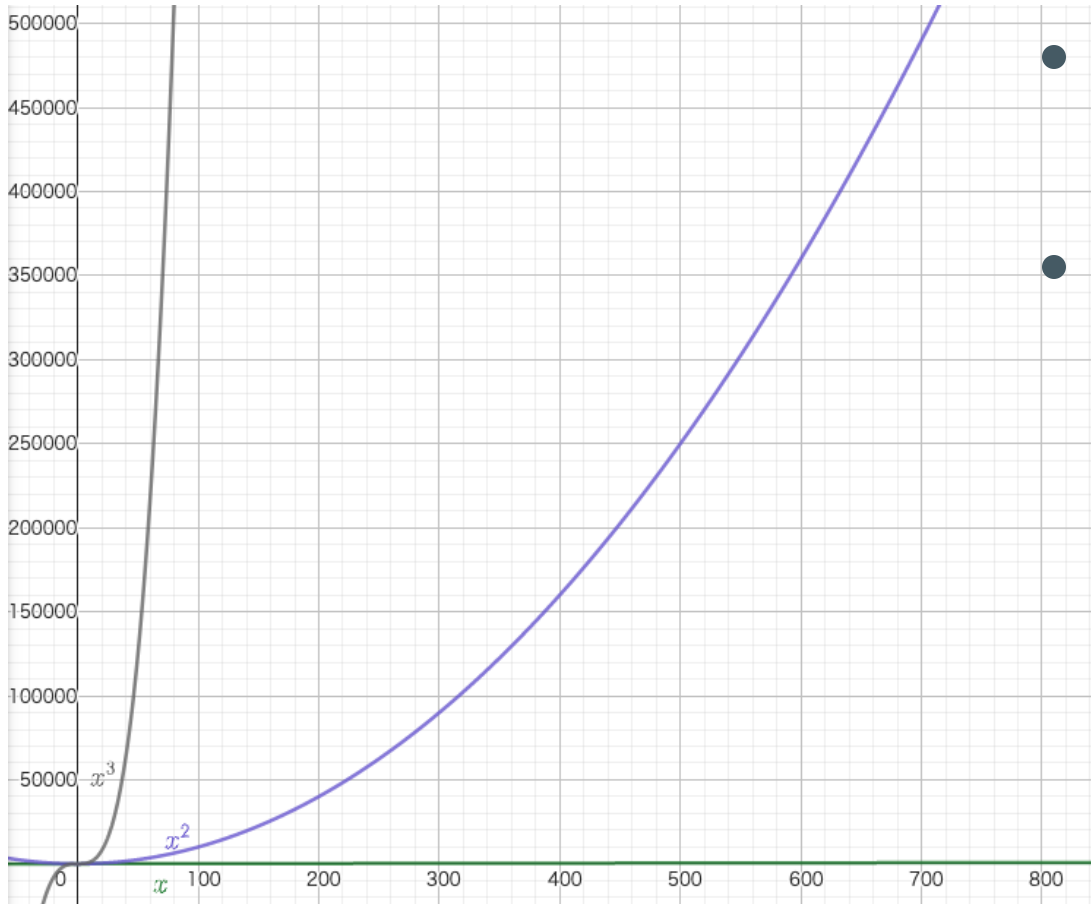
速いコードが書きたい！

でも速いコードってどうやって評価する??

- 「1,000,000個のデータに対して5秒で終了しました！」
 - データの個数が変わったらどうなる??
 - そもそもPythonで実行するかC言語で実行するかでも変わりそう

「データの大きさ」や「実行する環境」に依存しない評価方法が必要
→ 計算量の出番

オーダー記法 (1/2)



- n, n^2, n^3 では n が大きくなったとき値が大きく変化する
- 定数倍を考えないで、 n の項だけに注目すればいいのでは??

→ O (ランダウの記号) を用いて記述

オーダー記法 (2/2)

- 計算量は基本的にオーダー記法で書く
 1. 一番大きい項のみ残して表記する
$$n! > a^n > n^a > \log n > a \quad (a \text{ は定数})$$
 2. 定数倍は無視する

オーダー記法の例)

$$5n^3 + 4n^2 + 100n \longrightarrow O(n^3)$$

$$2^n + n^{100} + 10^9 n \longrightarrow O(2^n)$$

コードの計算量の調べ方

- n 回のループをする $\rightarrow O(n)$
- n 回のループの中で n 回のループをする（二重ループ）
 $\rightarrow O(n^2)$
- bit全探索（ n 個の要素についてある/ないの2通りを考える）
 $\rightarrow O(2^n)$
- n 個の順列を全て調べる $\rightarrow O(n!)$

問題！

このコードの計算量は？？（わかった人は高速化してみよう）

```
# 1~n までの数の和を求める
n = int(input())

ans = 0
for i in range(1, n+1):
    ans += i

print(ans)
```

答え

このコードの計算量は？？（わかった人は高速化してみよう）

```
# 1~n までの数の和を求める
n = int(input())

ans = 0
for i in range(1, n+1):
    ans += i

print(ans)
```

→ $O(n)$ (n までのループを1回している)

計算量を落とす

上のコードは、 $1 \sim n$ の和を求めるために $O(n)$ の計算をしています

↓この公式を使えば??

$$\sum_{i=1}^n = \frac{1}{2}n(n+1)$$

参考

- 計算量オーダーの求め方を総整理！ ～どこからlogが出て来るか～