

# Python Programming in Finance

盧政良

(Zheng-Liang Lu, Arthur)

台大資訊系

**本投影片將提供兩個重  
要的先備知識...**

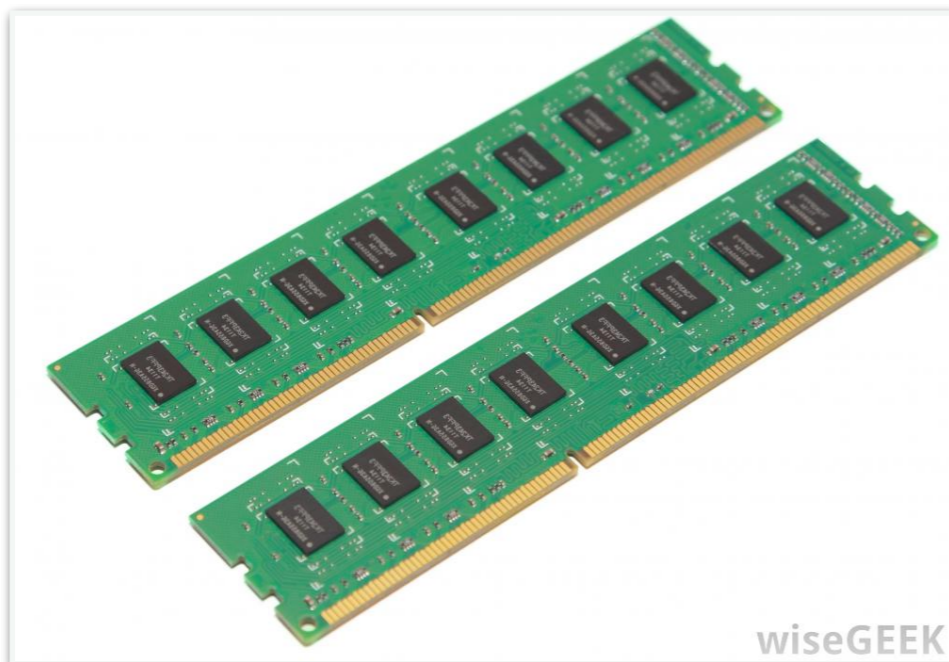
# 什麼是電腦？

[Inside your computer](#)

講古請參考：[Timeline of Computer History](#)

# CPU & Memory

- ▶ 算術邏輯單元  
*Arithmetic & Logic Units, ALUs*
- ▶ 暫存器  
*Registers*
- ▶ (還有很多)



- ▶ 記憶體

*Memory*



# 寫程式 = 教電腦解決問題

所以你得先知道怎麼做啊！ 

一隻腳踏在自己的專業領域，一隻腳踏在CS

# COMPUTATIONAL THINKING

## DECOMPOSITION

Breaking big problems into smaller, easier to manage problems



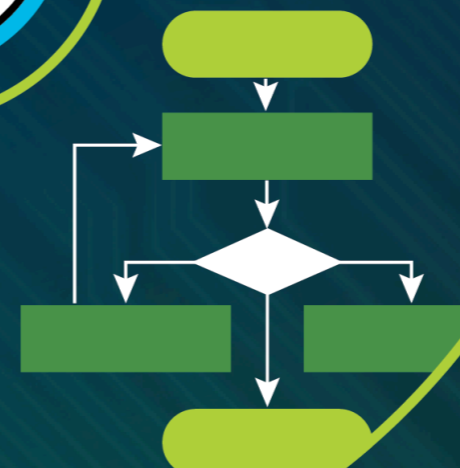
## PATTERN RECOGNITION

Analyze & look for a repeating sequence



Remove parts of a problem that are unnecessary and make one solution work for multiple problems

## ABSTRACTION

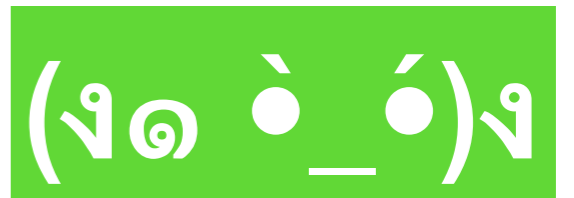


Step-by-Step instructions on how to do something

## ALGORITHM DESIGN

# 簡單的例子

- ▶ 存在一組資料，譬如： $\{4, -3, 9, 1, 8\}$ 。
- ▶ 請問這組資料的**最大值**是多少？



# 解題需要有策略

- ▶ 找別人幫忙寫？ 是在哈囉？
- ▶ 呼叫max之類的？
- ▶ 自己想辦法？
  - 想想看用眼睛看這筆數據的時候，大腦裡經歷了哪些步驟？

# 演算法

algorithm



<http://ed.ted.com/lessons/your-brain-can-solve-algorithms-david-j-malan>

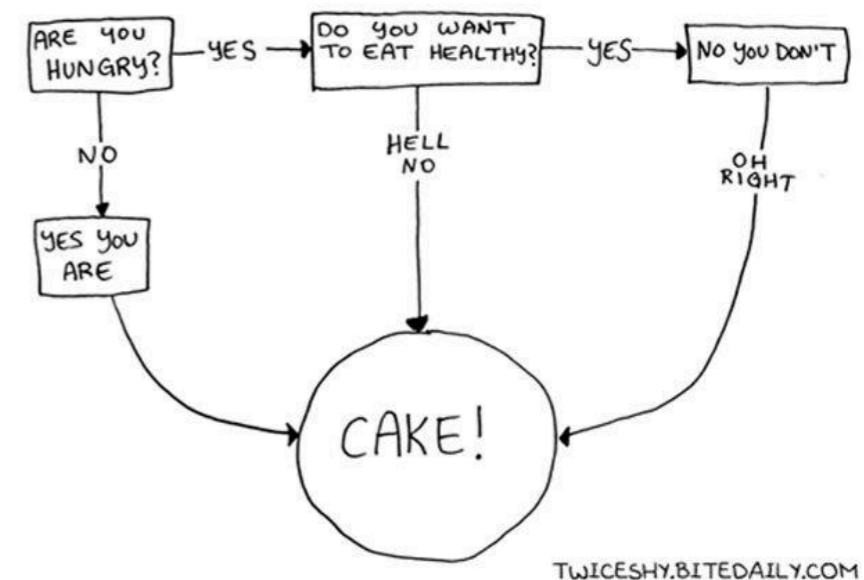
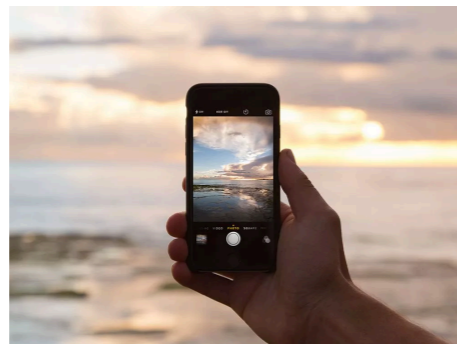
“Computers are good at following instructions, but not at reading your mind.”

– Donald Knuth (1938–present)

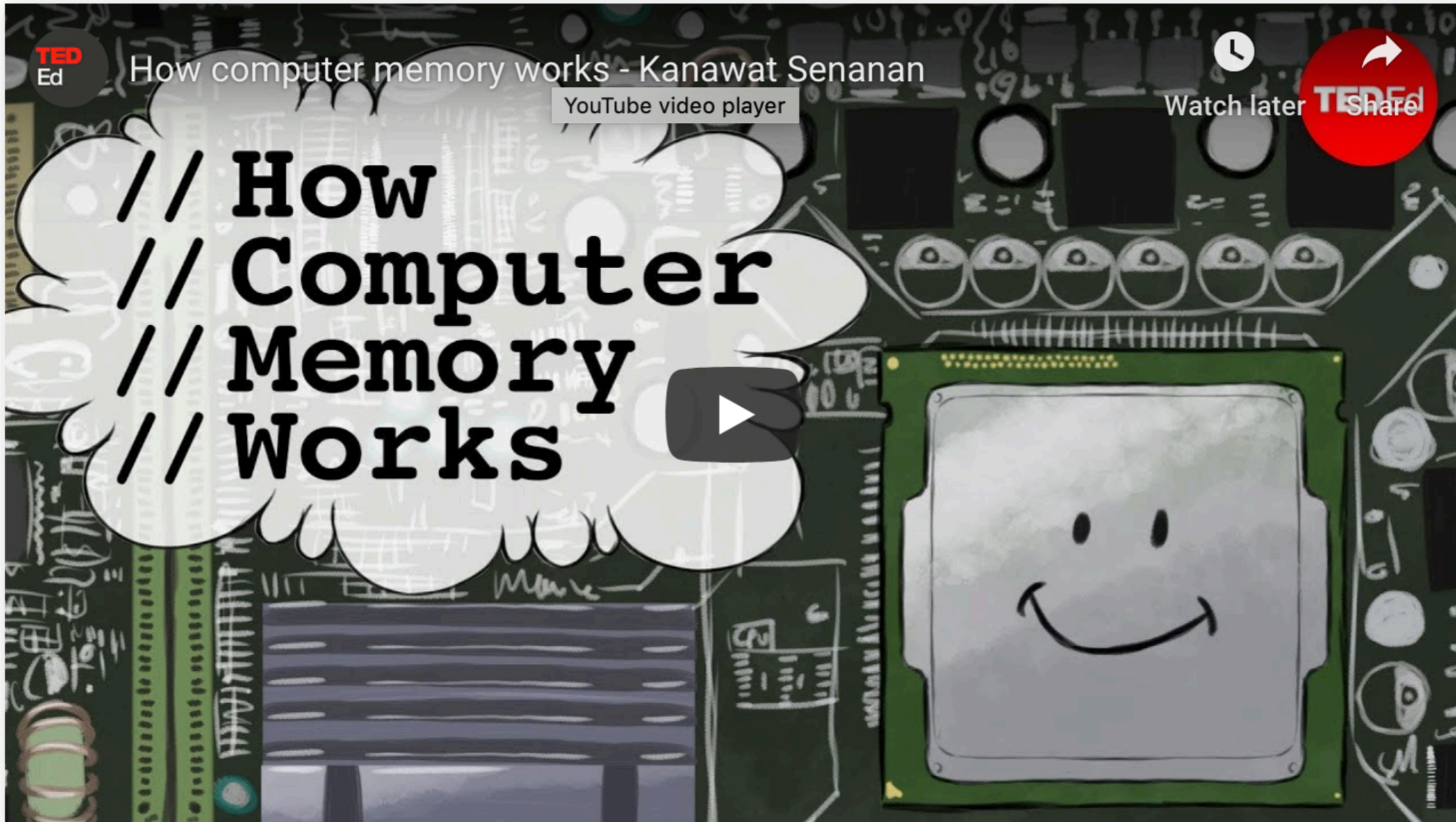


盧政良 台大資訊系

# 程式 = 資料 + 演算法







<https://ed.ted.com/lessons/how-computer-memory-works-kanawat-senanan>