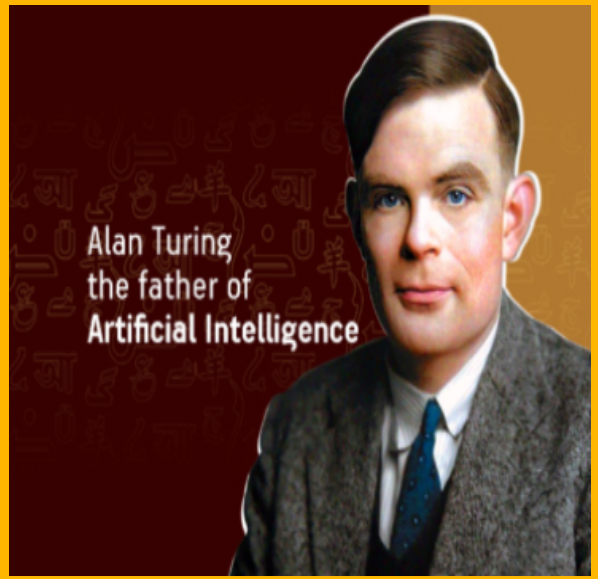



# Introduction to Computer Science Fall 2022 #12

Chi-Jen Wu

# 期中考成績

| 組界 | 人數 |
|---|---|
| 100-109 | 3 |
| 90-99 | 6 |
| 80-89 | 10 |
| 70-79 | 9 |
| 60-69 | 11 |
| 50-59 | 13 |
| 40-49 | 8 |
| 30-39 | 6 |
| 20-29 | 4 |
| 總計 | 70 |

平均：63.96

| 題目 | 人數 |
|---|---|
| 22 | 3 |
| 34 | 7 |
| 52 | 10 |
| 38 | 12 |
| 54 | 13 |
| 68 | 13 |
| 17 | 14 |
| 53 | 19 |
| 76 | 19 |

| 問答題 | 人數 |
|---|---|
| 19 | 9 |
| 9 | 7 |
| 2 | 3 |
| 5 | 3 |
| 17 | 0 |

22. A $(0111\ 1111)_2$ and B $(0000\ 0001)_2$ are stored in 2's complement notation. Show how A is added to B.

1. +130
2. -130
3. Error:overflow
4. -128
5. +128

34. The process A is reading data from the keyboard.
What is the status of the process A in each of the following situations?

1. Running
2. Ready
3. Waiting
4. Reading
5. Writing

**Eight-bit signed integers**

| Decimal value | Two's-complement representation |
|---|---|
| 0 | 0000 0000 |
| 1 | 0000 0001 |
| 2 | 0000 0010 |
| 126 | 0111 1110 |
| 127 | 0111 1111 |
| −128 | 1000 0000 |
| −127 | 1000 0001 |
| −126 | 1000 0010 |
| −2 | 1111 1110 |
| −1 | 1111 1111 |

52. In the curl Lab (Lab#4), when a user needs to retrieve a document from the server, and the SSL connection uses the _____ method in the HTTPS request.

1. SSL
2. SSH
3. DNS
4. TLS
5. HTTPS

38. Which of the following identifies the application to which a message arriving from the Internet should be given?

1. IP address
2. Port number
3. Mac address
4. DNS
5. HTTP

54. Which of the following statements is false for the Turing machine?

1. A problem that can not be solved by a Turing machine can not be solved by any algorithmic system
2. Turing machine is a conceptual device rather than an actual machine
3. A halting problem is unsolvable by any Turing machine
4. An NP problem is unsolvable by any Turing machine
5. None of the above

68. Which of the following protocol is considered as an application in network programming

1. UDP protocol
2. IP protocol
3. SMTP protocol
4. TCP protocol
5. ICMP protocol

17. The IEEE floating-point standard is
    1. IEEE 775
    2. IEEE 774
    3. IEEE 745
    4. IEEE 754
    5. IEEE 747

53. Which following sentence is not true for von Neumann architecture?
    1. It allows self-modifying code
    2. Memory holds both data and instructions
    3. CPU and memory are separated modules
    4. Most DSP processors use this architecture
    5. Invented by John von Neumann

76. In memory management, with the _____ technique, each program is entirely in memory and occupying contiguous locations; with the ____ technique, each program is still entirely in memory but not necessarily contiguous in memory; with the ____ technique, a program can be partly in memory.

1. pageing, demand paging, partitioning
2. paging, partitioning, demand paging
3. partitioning, paging, demand paging
4. partitioning, demand paging, paging
5. demand paging, partitioning, paging

17. Based on the following two tables (T1 and T2), interpret the result of the SQL query:

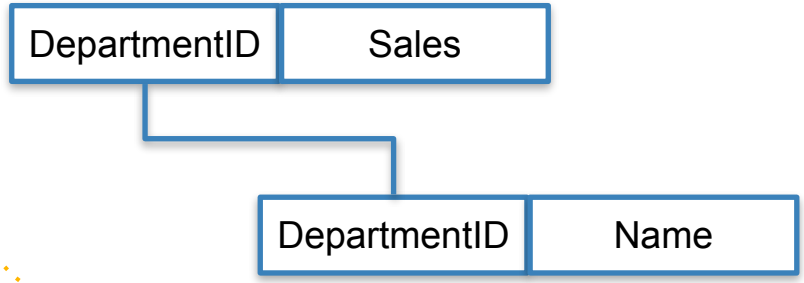
T1

| DepartmentID | Sales |
|---|---|
| 1 | 50 |
| 1 | 150 |
| 2 | 250 |
| 2 | 350 |

T2

| DepartmentID | Name |
|---|---|
| 1 | IT |
| 2 | HR |

Draw the schema of the two tables (T1 and T2)



17. Based on the following two tables (T1 and T2), interpret the result of the SQL query:

T1

| DepartmentID | Sales |
|---|---|
| 1 | 50 |
| 1 | 150 |
| 2 | 250 |
| 2 | 350 |

T2

| DepartmentID | Name |
|---|---|
| 1 | IT |
| 2 | HR |

2. SELECT
DepartmentID, sales, name
FROM T1, T2
WHERE T1.DepartmentID = T2.DepartmentID AND DepartmentID = '2'

第一筆： 2, 250, HR
第二筆： 2, 350, HR

**What is the von Neumann model, and what is the von Neumann bottleneck?**

CPU (Control unit, ALU), Memory, Input/output
And stored program concept

整個系統的速度終將受限於CPU和記憶體之間(匯流排 bus)的速度

**Write a SQL for adding a new student in the Student table (Name (“IU Lee”) and age (“26”).**

INSERT INTO Student VALUES (“IU Lee”, “26”);

**Explain the stored-program concept of the Von Neumann model.**

The data and programs should have the same format, and they are stored in memory

**What is the largest positive value that can be represented with a 2's complement 8 bit number?**
**What is the largest positive value that can be represented with an unsigned 8 bit number?**

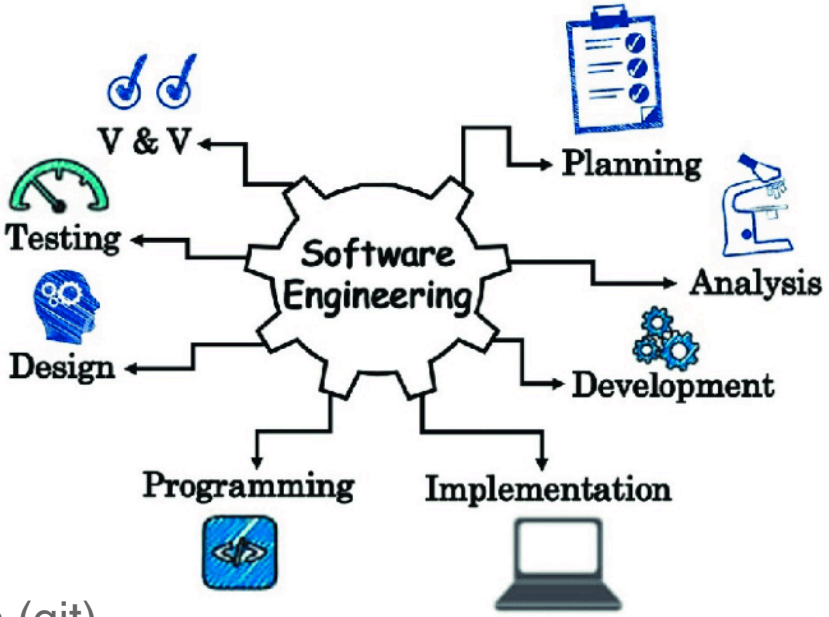
127
255

Eight-bit signed integers

| Decimal value | Two's-complement representation |
|---|---|
| 0 | 0000 0000 |
| 1 | 0000 0001 |
| 2 | 0000 0010 |
| 126 | 0111 1110 |
| 127 | 0111 1111 |
| −128 | 1000 0000 |
| −127 | 1000 0001 |
| −126 | 1000 0010 |
| −2 | 1111 1110 |
| −1 | 1111 1111 |

# Topics

- An Introduction to Computer Science
- The Shapes of Computers Today
- Computer Organization and Architecture
- Operating system
- Networking & The Internet
- Database Systems
- **Software Engineering**
- Cloud Platform/Cloud Shell Editor
- Cloud Platform/Cloud Source Repositories (git)
- C/C++ Programming



# Software Engineering

- **Why we need software engineering**
  - **An introduction**
- Software Life Cycle
- Methodologies
- Modularity
- Quality Assurance
- Documentation
- Current approaches
  - Code Review
  - **Coding style**

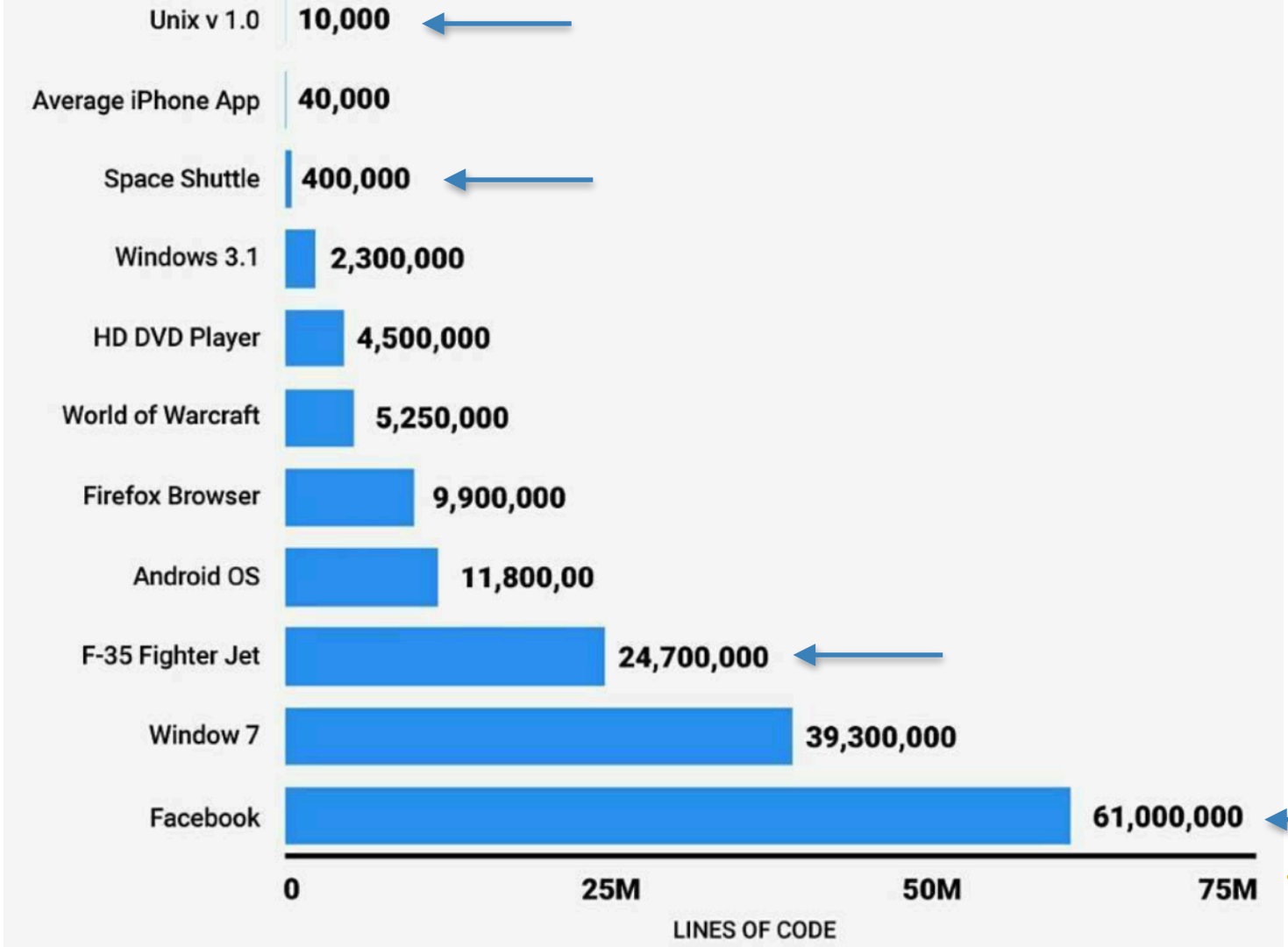
| | | |
|---|---|---|
| 資工系 | 3 | 軟體工程<br>Software Engineering<br>詳細資料 Detail |
| 資工系 | 3 | 網路應用軟體設計<br>Network Application Software Design<br>詳細資料 Detail |
| 資工系 | 3 | 物件導向軟體設計<br>Object-Oriented Software Design (OOSD)<br>詳細資料 Detail |
| 資工系 | 2 | 網頁程式設計<br>Web Programming<br>詳細資料 Detail |
| 資工系 | 2 | 資料庫系統設計<br>Database System Design<br>詳細資料 Detail |
| 資工系 | 3 | Unix程式設計<br>Programming in the Unix<br>詳細資料 Detail |

# Lines of code

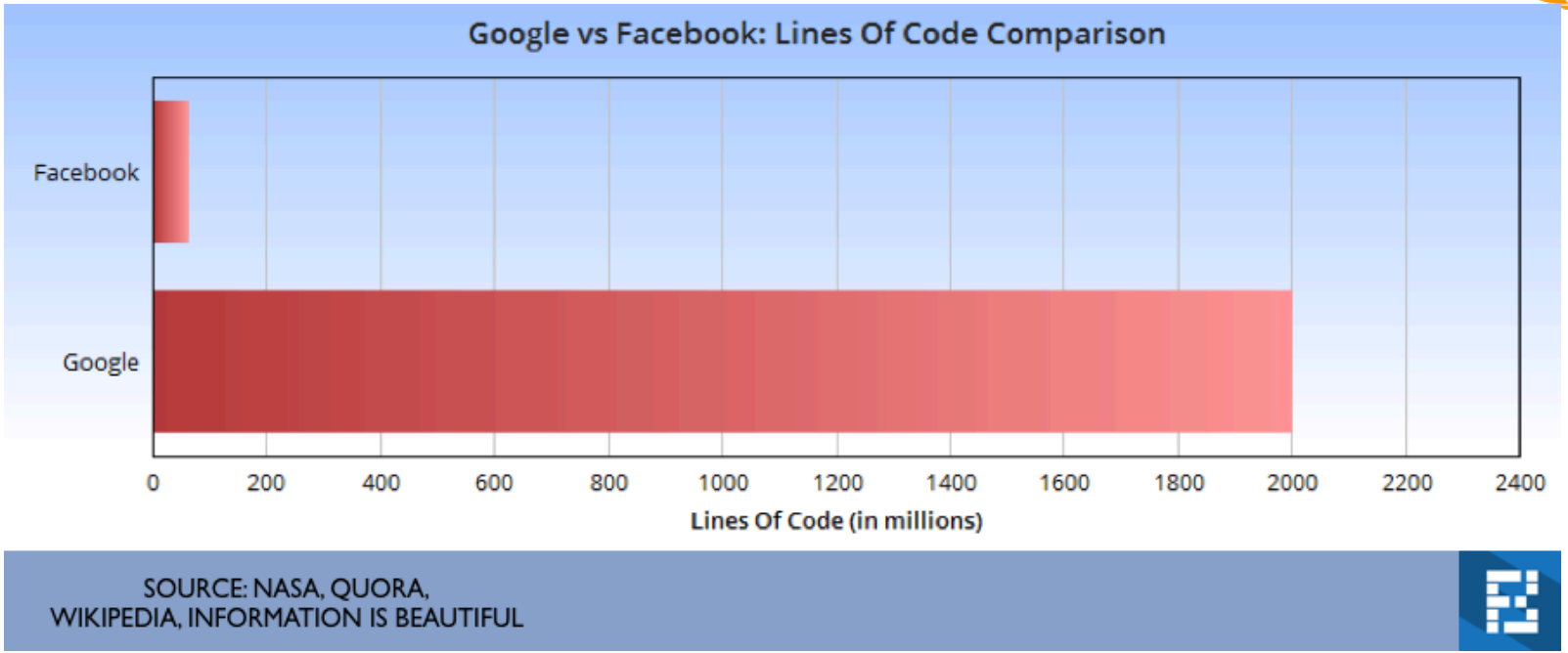
- 同學，資工系同學
- 你們覺得你自己可以處理多少行程式？
- 一千行？
- 一萬行？
- 十萬行？
- 五十萬行？
- 一百萬行？

# Lines of OS Code

| Operating System | Lines of Code |
| --- | --- |
| Windows | 89 million |
| Mac | 50 Million |
| Linux | 40 Million |


Unix v 1.0
10,000
Average iPhone App
40,000
Space Shuttle
400,000
Windows 3.1
2,300,000
HD DVD Player
4,500,000
World of Warcraft
5,250,000
Firefox Browser
9,900,000
Android OS
11,800,00
F-35 Fighter Jet
24,700,000
Window 7
39,300,000
Facebook
61,000,000
0
25M
50M
75M
LINES OF CODE

Google has **27,169** software engineers




https://www.informationisbeautiful.net/visualizations/million-lines-of-code/

# 一個五人到十人的團隊

- 約莫可以處理**十幾萬行**的程式碼
  - 大概就是極限了
- 隨著服務的增長
  - 舊功能要留
  - 新功能要加
  - 兩者也要能相容
  - **系統複雜度**就會爆炸性成長

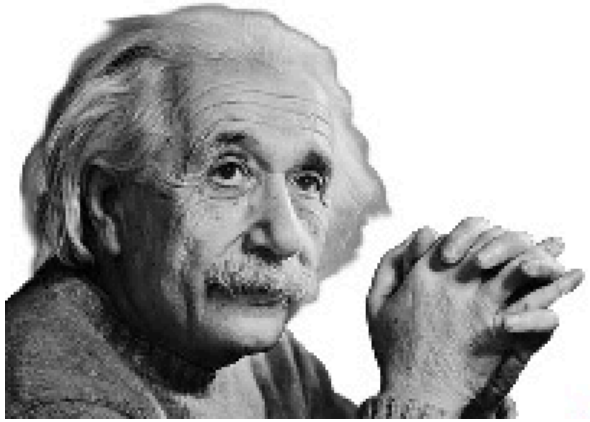
# 多複雜叫很複雜

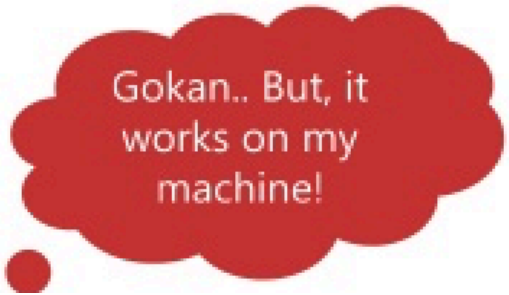
When things work, nobody knows why!

# First Law of Code Quality

$$\text{Error} = (\text{More Code})^2$$

$$E = MC^2$$

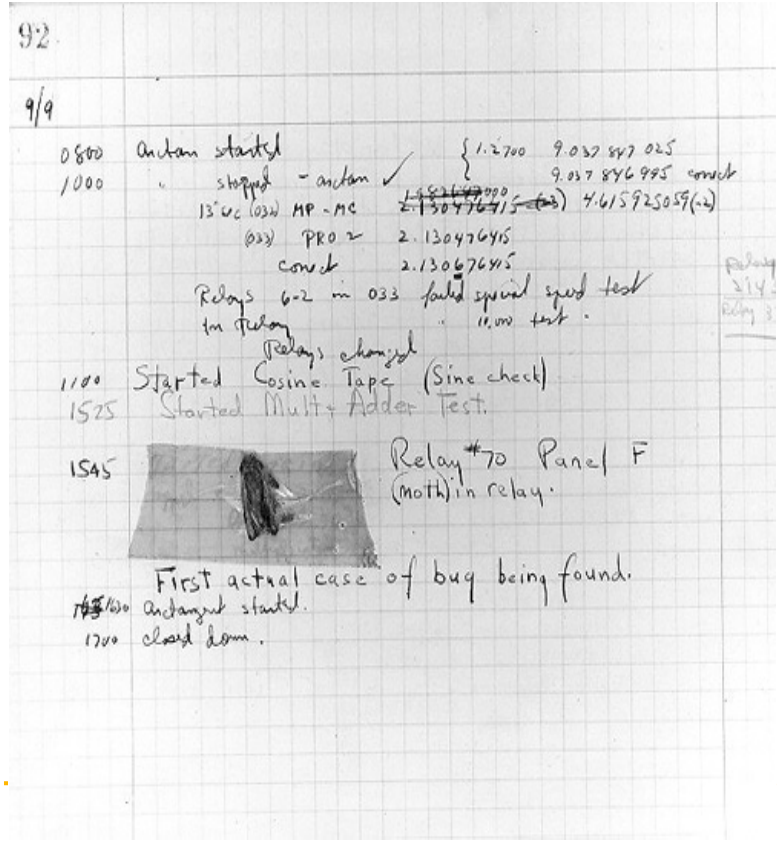




# The First Bug in CS

- Grace Hopper
  - Debug
  - COBOL
  - Y2K危機
    - 年月日各兩位
    - 120101
    - 990101

Grace Hopper, 1992



# Y2K Problem:
# 用兩個字元存西元19XX年

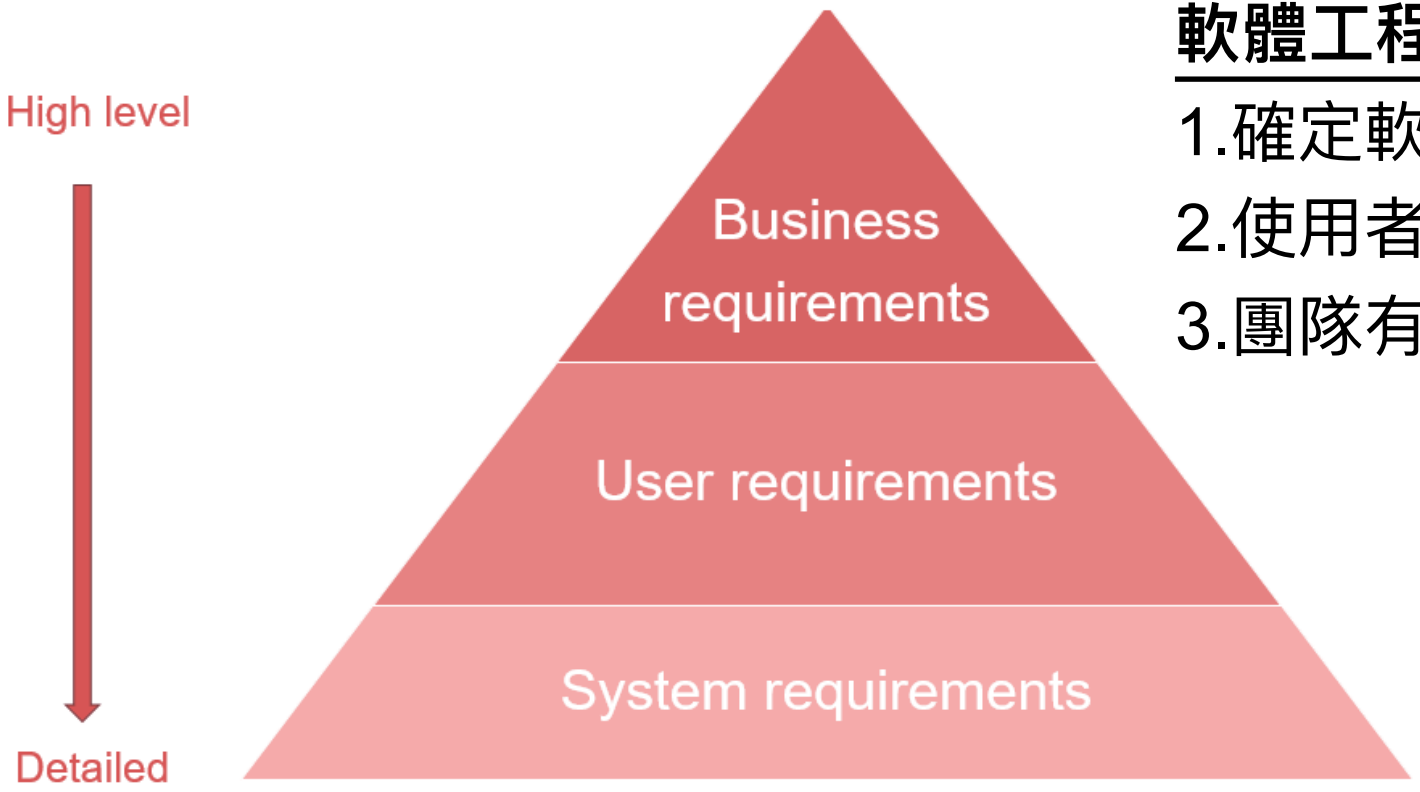
**設計程式的人**
**覺得2000年**
**就世界末日了**
**所以沒想過**
**西元2000年以後的事**


**這問題怎麼解決呢？**

# Software Engineering 包含

- Project planning
  - **Define your systems (problems)** **<— 最重要**
- Project management
- Documentation
- Prototyping and simulation
- Interface design
- **Programming** **<— 目前你最重要的事**

# Requirements classification



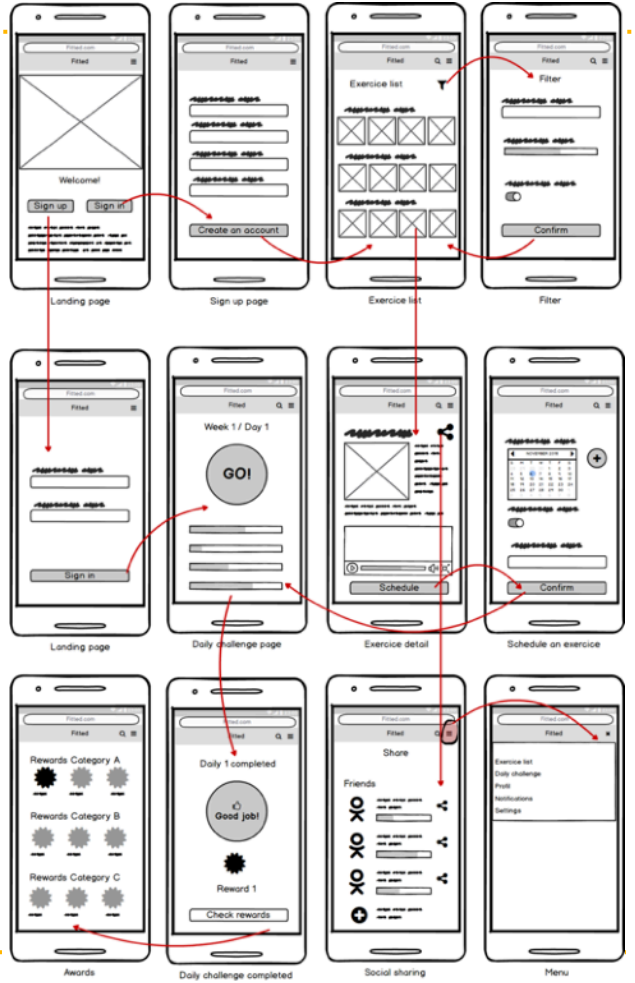

**軟體工程實戰**

1.確定軟體系統的目的

2.使用者是誰？

3.團隊有誰？

# Prototyping



# Current approaches

- 對於同學們目前最重要的兩件事
- **Code Review**
  - 吾自省吾身
- **Coding style**
  - 編寫原始碼的書寫風格
  - **易讀，別人也要看的懂**
  - **最好同一個團隊寫出來的程式碼，看不出來誰寫的**

計算機概論
課程下半
大一

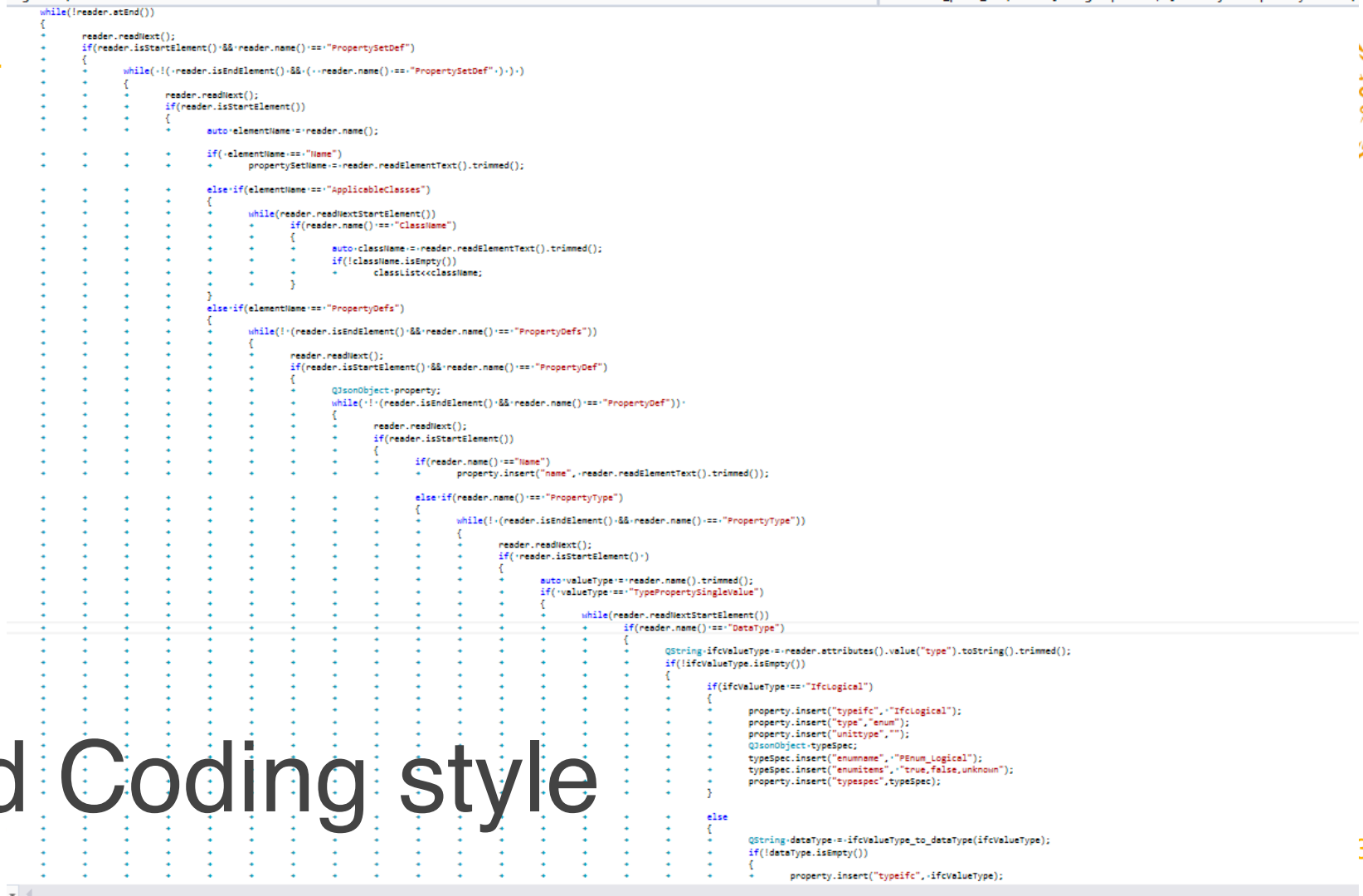
# Coding style

- **真的很重要！！**
- 在Google，所有的程式碼提交都需要經過批閱
  - Coding Style就是尤為重要的審閱內容之一
  - 包括每一個**空格**、**標點**、**對齊方式**，都必須遵循Google內部書寫代碼的**風格規則**

**面試官：程式碼如屎的豬隊，我不要！**

# Bad Coding Style

# Bad Coding style




# Bad Coding style

# Bad Coding style

```
/**
 * returns the most frequently occurring number of the set, or a random number from those provided
 */
protected int selectModeOrRandom(int a, int b, int c, int d)
{
    return b == c && c == d
            ? b : (a == b && a == c
                ? a : (a == b && a == d
                    ? a : (a == c && a == d
                        ? a : (a == b && c != d
                            ? a : (a == c && b != d
                                ? a : (a == d && b != c
                                    ? a : (b == c && a != d
                                        ? b : (b == d && a != c
                                            ? b : (c == d && a != b
                                                ? c : this.selectRandom(new int[] {a, b, c, d}))))))))));
}
```

# Poor coding style examples

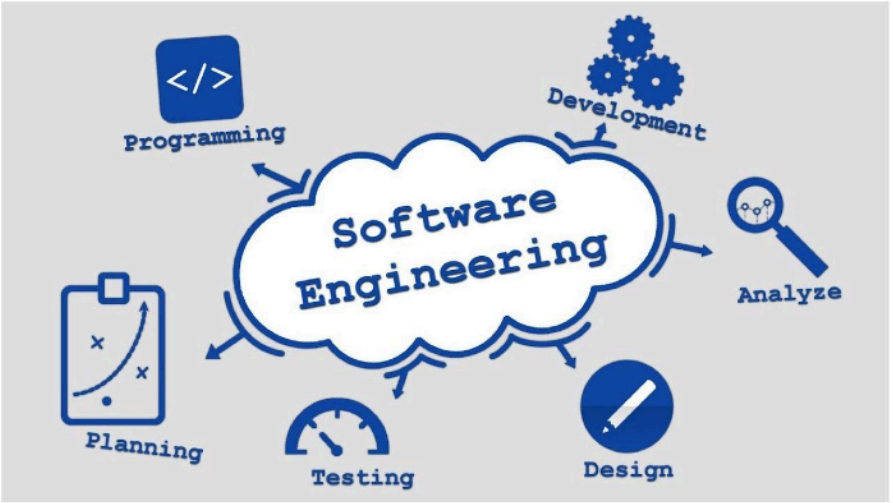
- Bad indentation　最基本的要求：縮排風格 (Indentation style)

```
if (a == b) {
while (c > d) {
if (c == a) {
...
}
else {
if (bool) {
call(b);
}
}
}
doSomething();
}
```

```
if (a == b) {
   while (c > d) {
      if (c == a) {
            ...
      }
      else {
            if (bool) {
                  call(b);
            }
      }
      doSomething();
   }
}
```

# Conclusion

- Why we need software engineering
  - An introduction
- Software Life Cycle
- Methodologies
- Modularity
- Quality Assurance
- Documentation
- Current approaches
  - **Code Review**
  - **Coding style**



# Thanks!

## Open for any questions

**CJ Wu**

cjwu@mail.cgu.edu.tw