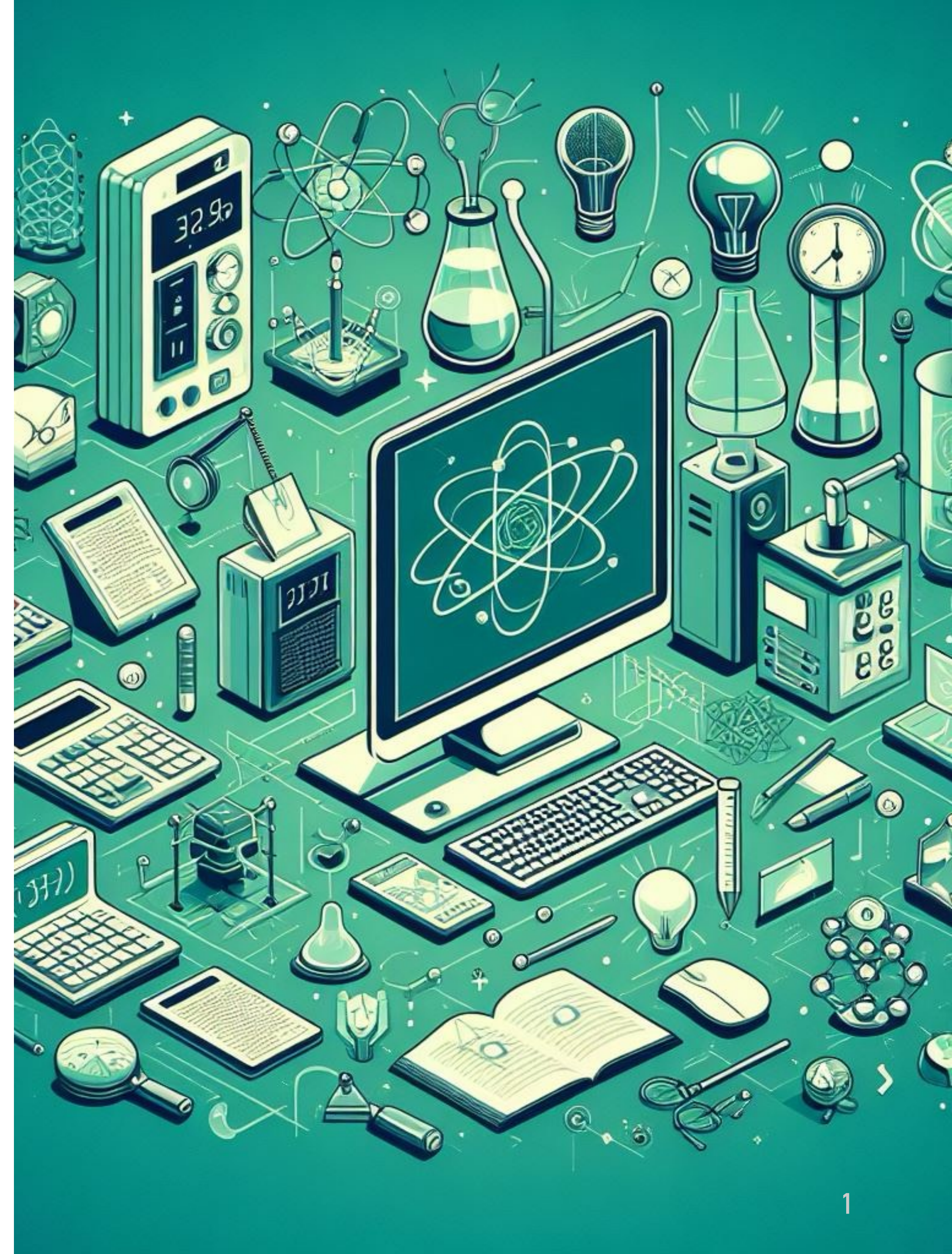


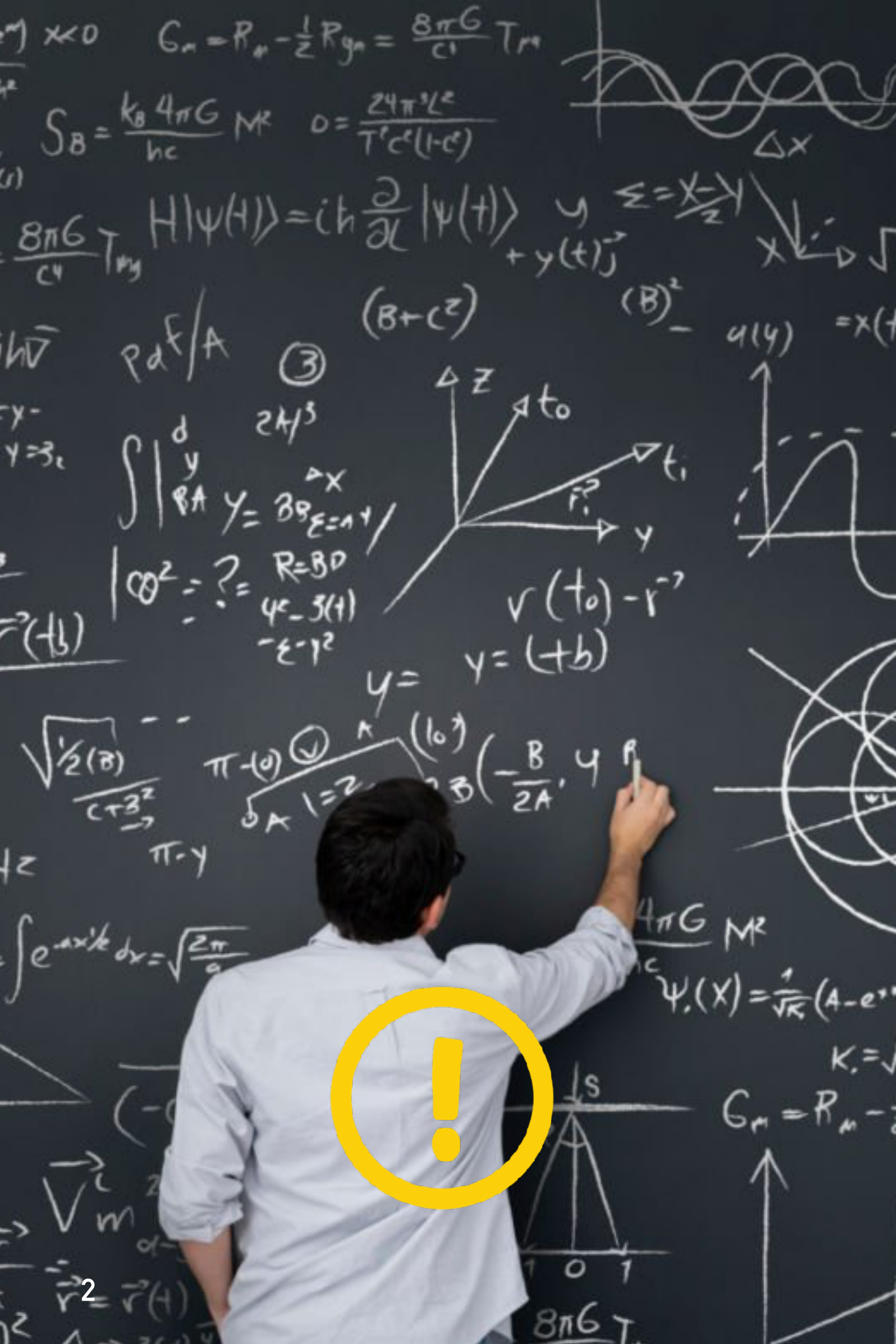
self-learning
✓

A COMPUTING COURSE FOR FUTURE PHYSICISTS

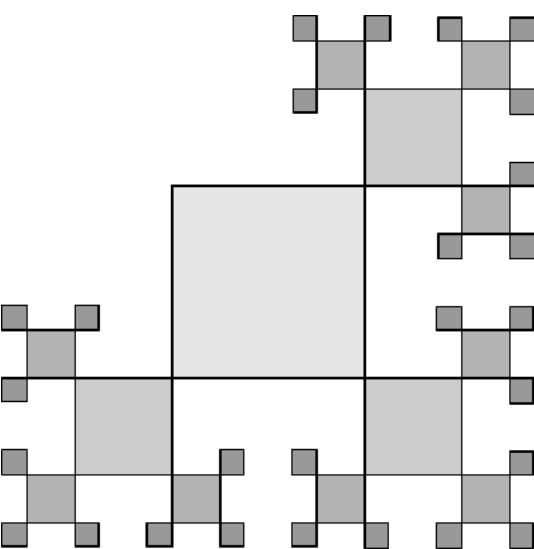
給未來物理學家的電腦課

臺大物理系 陳凱風



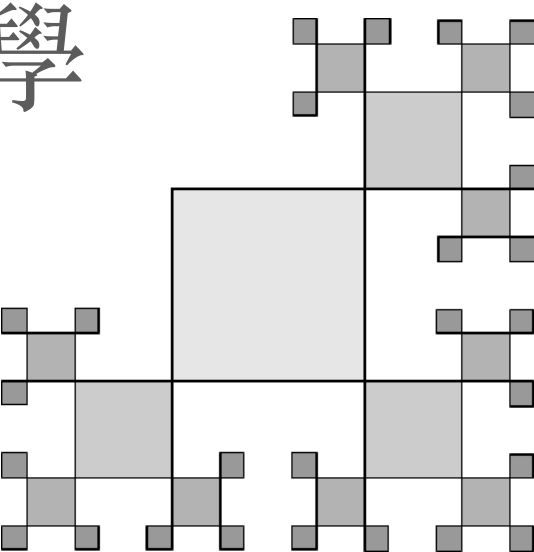


至今幾乎不可能不倚
賴電腦計算來解決現
代物理問題了！



一點點背景緣由

- ❖ 台大物理系準備了數個電腦計算相關課程：
 - 趙挺偉老師：「**Computational Physics / 計算物理**」（博班選修、半年三學分；以演算法為主，討論物理問題以及方法、如Lattice QCD等等；後來加上GPU加速計算等等技術)
 - 石明豐老師：「**Programming & Physics Simulation / 程式設計與物理模擬**」（大學部配合普物教材、半年一學分；VPython模擬、以專題自學、專注在模擬物理現象)
 - 高英哲老師：「**Programming & Numerical Analysis / 程式設計與數值分析**」（大學部必修、半年三學分；程式語言python教學、數值方法)
- ❖ 在2014從英哲老師繼承「程式設計與數值分析」課程。2018年程式設計拿掉並改制成「**Introduction to Numerical Analysis / 數值分析導論**」（大學部選修、半年三學分；以數值方法為主，並加上一些特殊題材，如機器學習、統計方法等）。



大學部計算課程強化

- ❖ 2020年之後，為了強化大學部電腦計算課程，在課程委員會協調下將「數值分析導論」重新架構成一年課程。並與系學會討論、盡量滿足不同需求；同時因為COVID疫情肆虐，必須能完全線上上課。
- ❖ 因此準備了「**Introduction to Computational Physics / 計算物理導論**」的這門課：
 - 大學部選修、兩學期共六學分。
 - **可以完全線上上課、翻轉教學設計**
 - 學習如何用電腦而不是用筆和紙解決科學問題。
 - 了解如何利用現有的工具/功能，或建立自己的工具。
- ❖ 本來課程標題想要設定成「**A Computing Course for Future Physicists / 給未來物理學家的電腦課**」，但是被系學會同學投了否決票...



課程設計：宴席餐會VS自助餐

- ❖ 傳統課程/傳統宴席雖然吃的豪華，可是難以符合所有人的需求...
- ❖ 自助餐設計：**自己決定菜單！**

豉油炆豆
紹酒燻筍
粉蒸牛肉
豉汁蒸魚
香雞
蒜泥白肉
酸辣鮮魚湯
六一式
蔥炆烏參
南腿白菜
干煸好味
蠔油炆肉
西瓜鐘



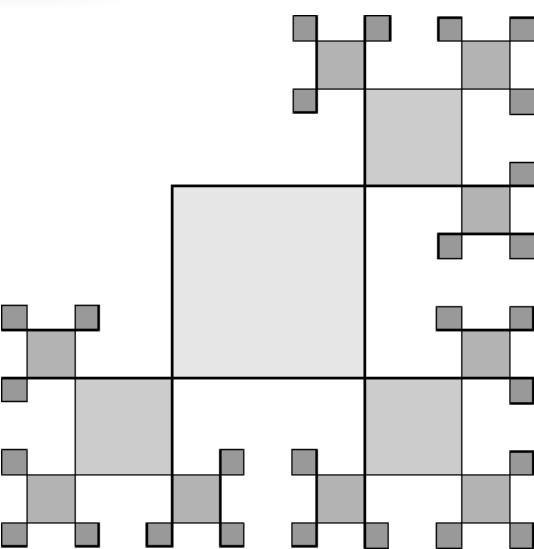
這門課基本上設計成「自助餐」，選擇自己需要的內容就好！

課程設計：上課學習VS電玩遊戲

- ❖ 為何上課學習是枯燥的、電玩遊戲卻是吸引人的？即使現代遊戲的複雜度常常不亞於專業知識...?
- ❖ 上課學習的反饋時程長，作業與考試難以快速確認作答正確與否、也不能再次嘗試。電玩則相反，可以重複嘗試與挑戰。



所以能夠「重複嘗試」、**「快速反饋」** 是一大重點。



課程實際安排

- ❖ **翻轉教學**：所有課程皆預錄、分成72個模組；每個模組包括投影片、講解錄影、範例程式、作業；要求學生自行消化內容並寫作業、但不需要全部完成。
- ❖ **所有預錄材料整合至單一網頁，所有的作業都可以自動批改、整合成績。**
- ❖ 因應學生需求同時使用python與C++為基礎語言，也不需要都學會。部分作業(數值方法、計算物理)的部分可以使用雙語作答。
- ❖ 內容包括：python、C++、數值方法、計算物理專題、機器學習專題、~~統計專題~~。
缺習題無開放
- ❖ 上課時間活動：協助疑難雜症解決與套件安裝(兩週)、TA提供練習題(五週)、期中口頭報告(7周)、期末競賽、期末海報報告。

課程使用
套件



python



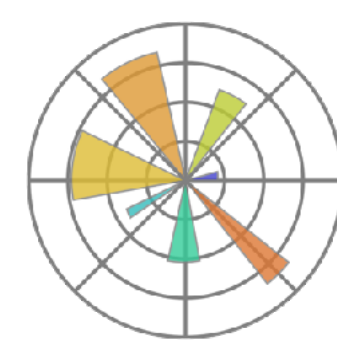
C++



NumPy



SciPy



Matplotlib



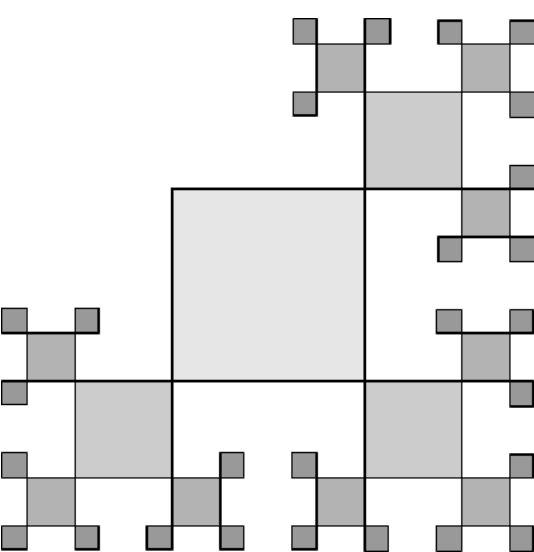
ROOT



GSL



Tensorflow

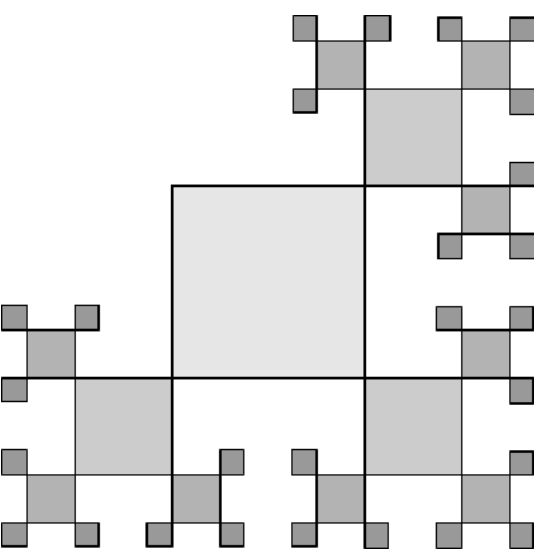


課程整合網頁: KAKIX

The screenshot shows the KAKIX website interface. On the left, there is a sidebar with a logo, navigation buttons for 'SANDBOX', 'EXPLORE', and 'SUMMARY', and a user profile for 'kfjack Kai-Feng Chen' with a 'Logout' button. The main content area has a dark blue header with the word 'EXPLORE'. Below the header, it says 'PICK UP YOUR MODULE AND PROCEED!' and 'You can find the categories of modules given below!'. A paragraph of text describes the categories: BASIS, C/C++, PYTHON, NUMERICAL, PHYSICS TOPICS, MACHINE LEARNING, and STATISTICS. Below the text is a diagram of course modules represented as hexagons. The 'Basis' module includes 'About This Course', 'Conceptual Fun Demos', 'Problem Solving Tips', 'Basic Unix Commands', and 'Introduction to Git'. The 'C/C++ Part I' module includes 'Into the C/C++ Language', 'Basic Elements I', 'Basic Elements II', 'Functions', 'Code Formatting I', 'Code Formatting II', 'Fundamental Data Types', 'C/C++ Operators', 'Object Scope & Conversions', and 'Control Flow'. The 'C/C++ Part II' module includes 'Compound Types & Array', 'Pointer & Reference', 'Memory Allocation & More on Arrays', 'More on Functions', 'Operator Overloading', 'Object Inheritance', 'Templates', 'I/O & Error Handling', and 'Object-Oriented Programming'. The 'Object-Oriented Programming' hexagon is circled in red.

- ❖ 整合上課內容、作業、成績。大幅降低TA工作，**不用消耗大量TA批改作業**。
- ❖ 因為內容特殊(數值分析、計算物理題目)，難以使用現成系統。
- ❖ 倚賴自行架設的Linux伺服器 + 自行開發的網頁(伺服器端：php+python，用戶端：javascript)。


一格「蜂巢」就是一個課程模組



課程模組

成績總覽

作業問題



The sidebar contains the following elements from top to bottom:

- KaKiX logo
- SANDBOX button
- EXPLORE button
- SUMMARY button
- User profile for kfjack (Kai-Feng Chen) with a Logout button
- Navigation buttons: PREV MODULE, INTRO, EXERCISE #1 (checked), EXERCISE #2 (checked), NEXT MODULE
- Module progress indicator: Module T2, 2/3 accomplished

MODULE T2 » NONLINEAR DYNAMICS I

Fresh before
Fri, 16 Jun 2023 13:00

NONLINEAR DYNAMICS I

困難等級

「賞味期限」

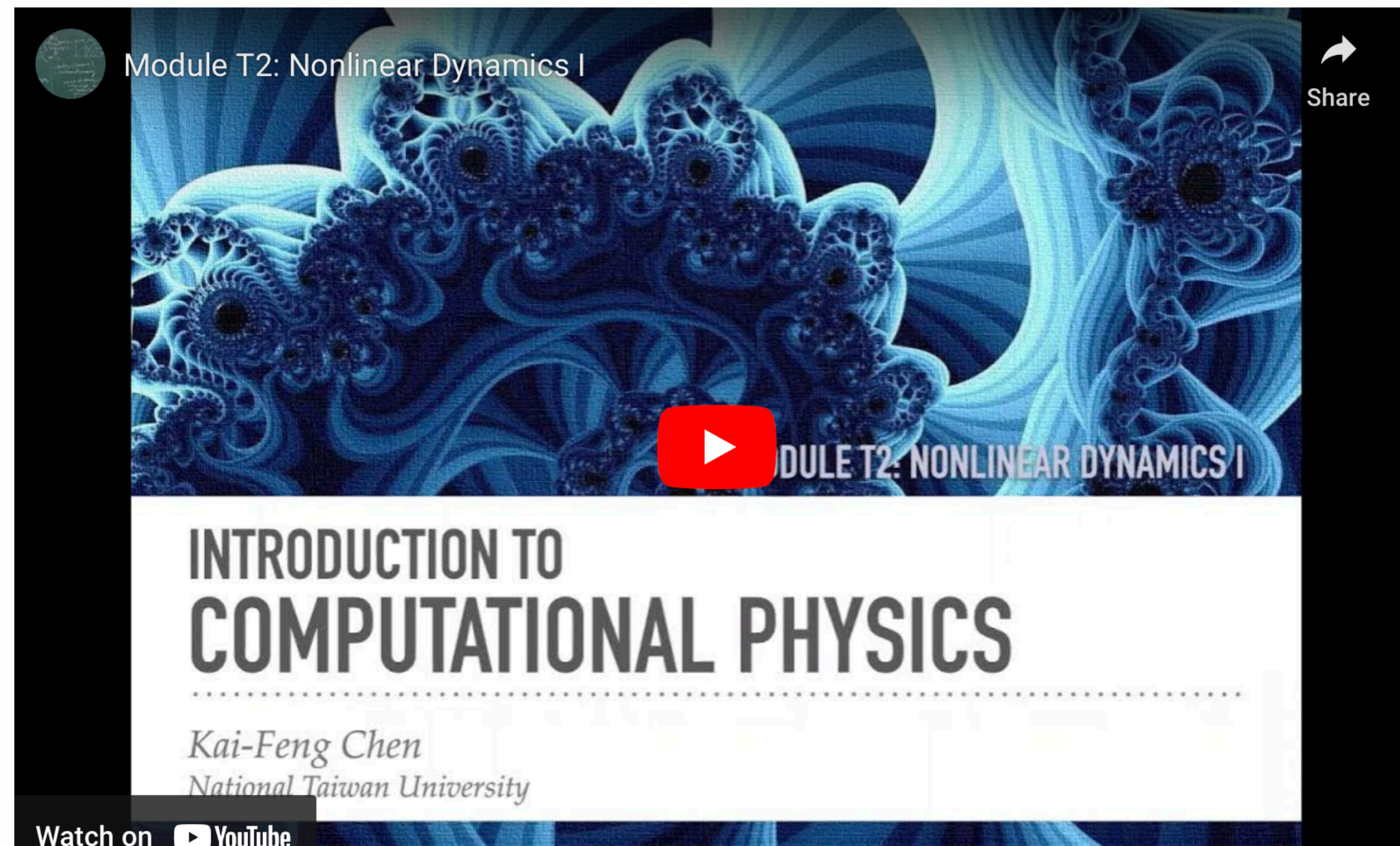
In this module we are going to introduce the logistic map, as a classical example with nonlinear dynamics. The map could have a fixed point value, or multiple attractors, or in the end, becomes chaotic.

Module Materials

- slides-mt2.pdf
- ex_mt2_1.py ex_mt2_1.cc
- ex_mt2_2.py ex_mt2_2.cc
- ex_mt2_3.py ex_mt2_3.cc
- ex_mt2_4.py ex_mt2_4.cc
- ex_mt2_4a.py ex_mt2_4a.cc
- ex_mt2_5.py ex_mt2_5.cc

投影片/範例程式

Recorded Video



The video player shows a thumbnail with a fractal background. The text on the thumbnail includes: 'Module T2: Nonlinear Dynamics I', 'INTRODUCTION TO COMPUTATIONAL PHYSICS', 'Kai-Feng Chen', and 'National Taiwan University'. A red play button is centered over the thumbnail. A 'Share' button is in the top right corner. At the bottom, it says 'Watch on YouTube'.

課程錄影

- ❖ 每個模組包括投影片、講解錄影、範例程式、作業。
- ❖ 當學生完成模組的全部作業可以獲得「金幣」。

練習問題

問題敘述

選擇語言(C++ or Python)

MODULE T2 » EXERCISE #1

Fresh before
Fri, 16 Jun 2023 13:00

ECOLOGY MAP #1: BIFURCATION DIAGRAM

The ecology map is a typical map which can lead to non-linear behaviors. The map, which has a definite relationship between x_i and x_{i+1} steps:

$$x_{i+1} = x_i e^{\mu(1-x_i)}$$

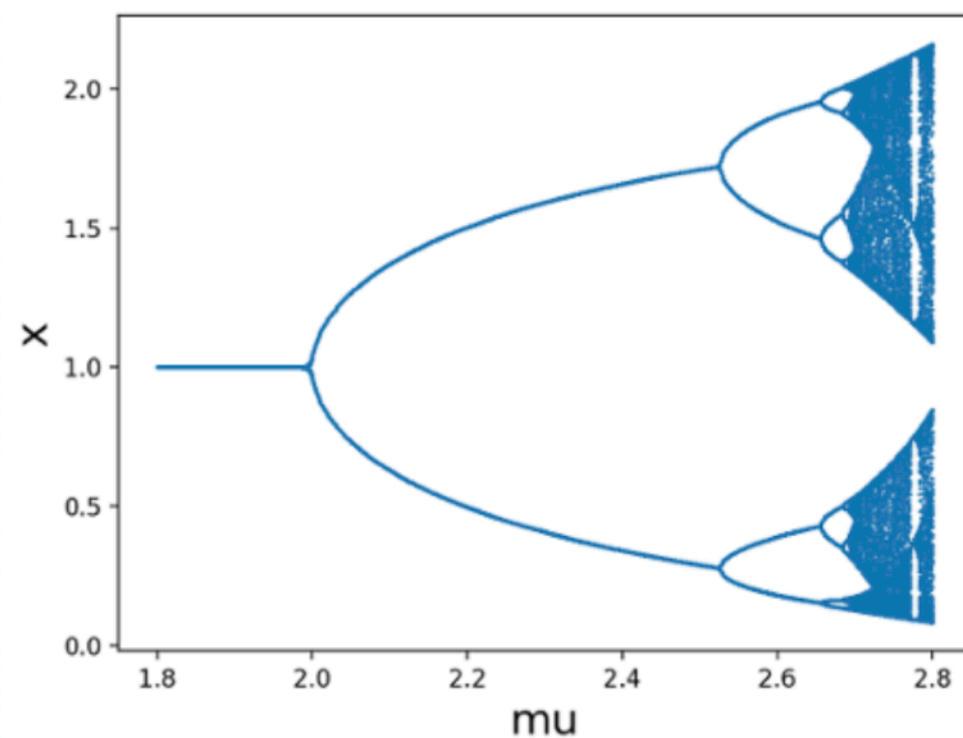
where μ is a tunable parameter. The behavior of ecology map depends on the value of the parameter μ . For example, if $\mu = 1.8$, the map will converge to single fixed point; if $\mu = 2.3$, the map will end up with two attractors; if $\mu = 2.6$, the map will end up with four attractors, and become chaotic if $\mu = 2.9$.

In order to study such a behavior of population increasing, it is helpful to examine the **bifurcation diagram**, which consists of a plot of attractors versus the growth parameter μ . Please implement a function which takes no parameter, but simply returns the bifurcation diagram as an **array of 8,000 elements**. The bifurcation diagram should be generated with the following steps explicitly:

- Starting from $\mu = 1.80$, in the spacing of 0.01 for 80 steps. This means all the trial μ values are 1.80, 1.81, 1.82, ..., 2.58, 2.59.
- For each trial μ value, generate the x series with an initial value of $x_0 = 0.5$. As an example for $\mu = 1.80$, the first few values in the series are $x_0 = 0.5$, $x_1 = 1.2298$, $x_2 = 0.8132$, $x_3 = 1.1382$, ...
- For each trial μ value, store the values of x_{201} , x_{202} , ..., x_{300} in the resulting array. This means the resulting array of 8,000 elements are simply given by

$$\begin{aligned}
 &x_{201}^{\mu=1.80}, x_{202}^{\mu=1.80}, x_{203}^{\mu=1.80}, \dots, x_{300}^{\mu=1.80}, \\
 &x_{201}^{\mu=1.81}, x_{202}^{\mu=1.81}, x_{203}^{\mu=1.81}, \dots, x_{300}^{\mu=1.81}, \\
 &x_{201}^{\mu=1.82}, x_{202}^{\mu=1.82}, x_{203}^{\mu=1.82}, \dots, x_{300}^{\mu=1.82}, \\
 &\dots \\
 &x_{201}^{\mu=2.59}, x_{202}^{\mu=2.59}, x_{203}^{\mu=2.59}, \dots, x_{300}^{\mu=2.59}
 \end{aligned}$$

The resulting bifurcation diagram should look like the following figure (between $\mu=1.8$ and 2.6):



作答輸出

Language Mode

C/C++ Python

Your Code

XL L M S RESET

```

1 #include <vector>
2
3 std::vector<double> bifurcations_ecology_map() {
4     std::vector<double> data(8000);
5     // START YOUR CODE HERE //
6
7     data.clear();
8     double mu = 1.80;
9     while(mu<2.595) {
10        double x = 0.5;
11        for(int i=1;i<=300;++i) {
12            x *= exp(mu*(1.-x));
13            if (i>=201 && i<=300)
14                data.push_back(x);
15        }
16        mu += 0.01;
17    }
18
19    // END YOUR CODE HERE //
20    return data;
21 }

```

作答區

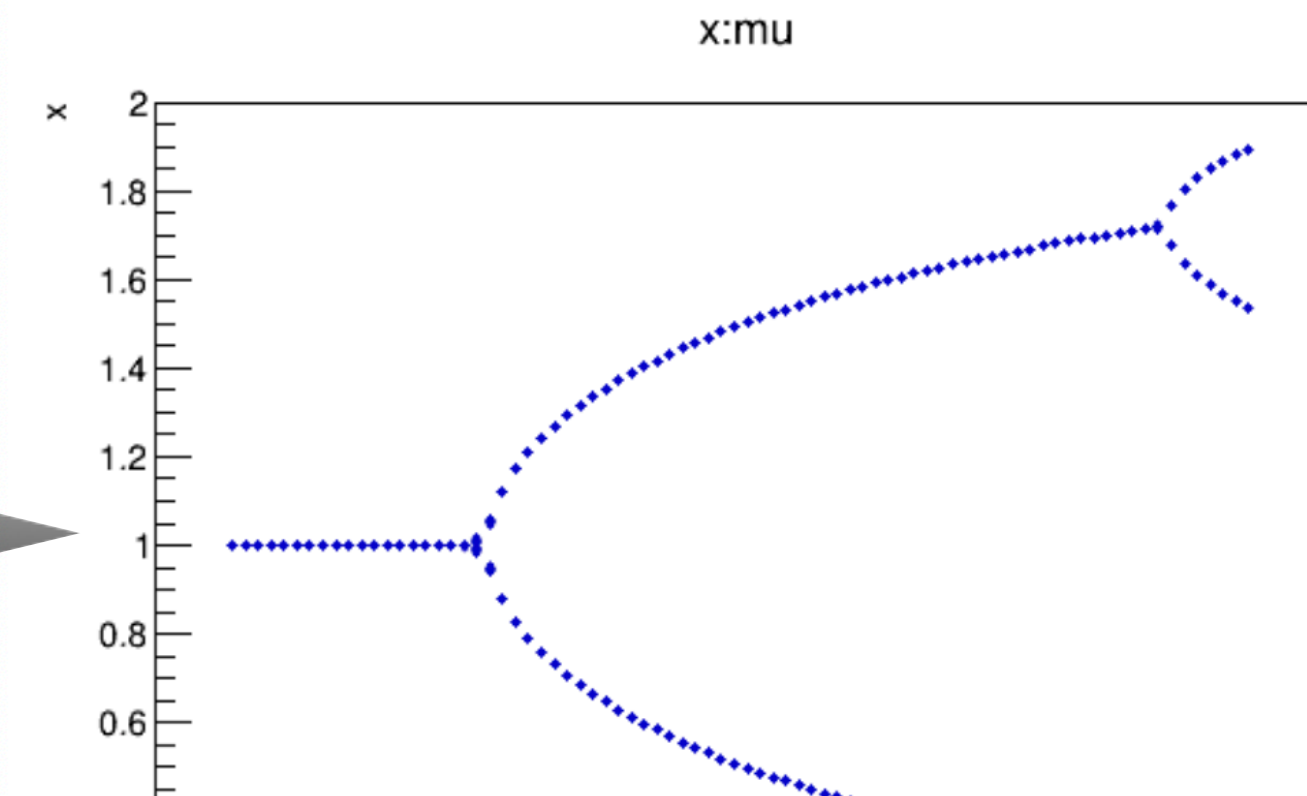
Submit

上傳答案 + 檢查



Success

Output:



如果答錯的話...



Failed
incorrect result



Exception
CPU time ran out



Warning
preserved code overwritten

DEMO: 課程網頁 × 課程模組 × 練習問題

DEMO: 課程教材 @ Backup



Please enter your account and password:

Account:

Password:

Login Set password

登入

選擇模組

消化教材

寫練習題

...

成績整合、總覽

- ❖ 為了讓上課更像遊戲一點，學生的成績可以在總覽直接看到，並提示還需要多少的努力才可以提升等第。
- ❖ 基礎成績來自作業的「金幣」，但是有上限（最多能只能加等第到B+）。
- ❖ 每完成一整個模組就可以根據難度獲得對應金幣，但是超過「賞味期限」的話就會減少獲得的金幣量。
- ❖ 一部分模組到了下學期就會完全過期，被設定成「化石」，金幣獲得量就變得很少，鼓勵學生推進進度。

要拿到高分，還需要收集
「獎牌」「獎盃」

SUMMARY

SUMMARY OF YOUR PROGRESS!

Collected Achievements

 **211 COINS**

 **MEDAL × 0**

 **TROPHY × 0**

B+

Toward next rank: win 8 medals or 1 trophy

Modules Progress Summary

吸引學生互動、參與課堂活動



「獎牌」 等於上課互動

- ❖ 獲得八面獎牌 = 多一個成績等第
- ❖ 參與TA提供的課堂習題 (最多獲得獎牌5次)。
- ❖ 替其他同學的期中報告提問與評論 (最多獲得獎牌7次)。



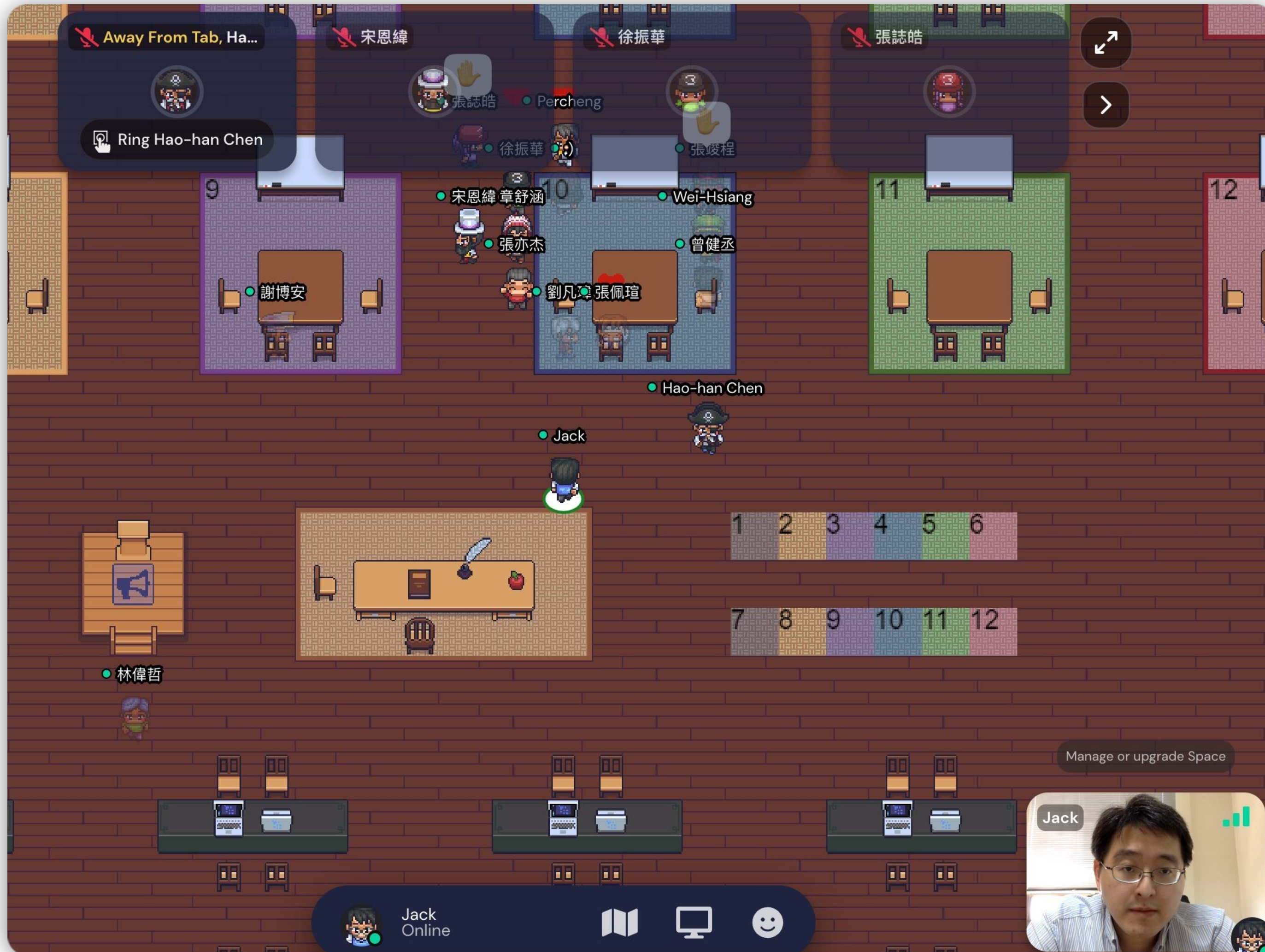
「獎盃」 等於課堂參與

- ❖ 獲得一個獎盃 = 多一個成績等第
- ❖ 參與期中口頭報告 (限制延伸上課相關主題) 可獲得一個。
- ❖ 參與期末海報 (任意主題) 可獲得一個。
- ❖ 參與期末遊戲競賽打敗半數同學可獲得一個。
- ❖ 設計新的課程模組可獲得一個。

所有活動都是自由參加，不參加也可以；但是想要好成績還是要參與。**想拿到A+一定拿得到。**

線上虛擬教室互動

❖ 疫情期間使用Gather.Town建立的虛擬教室：

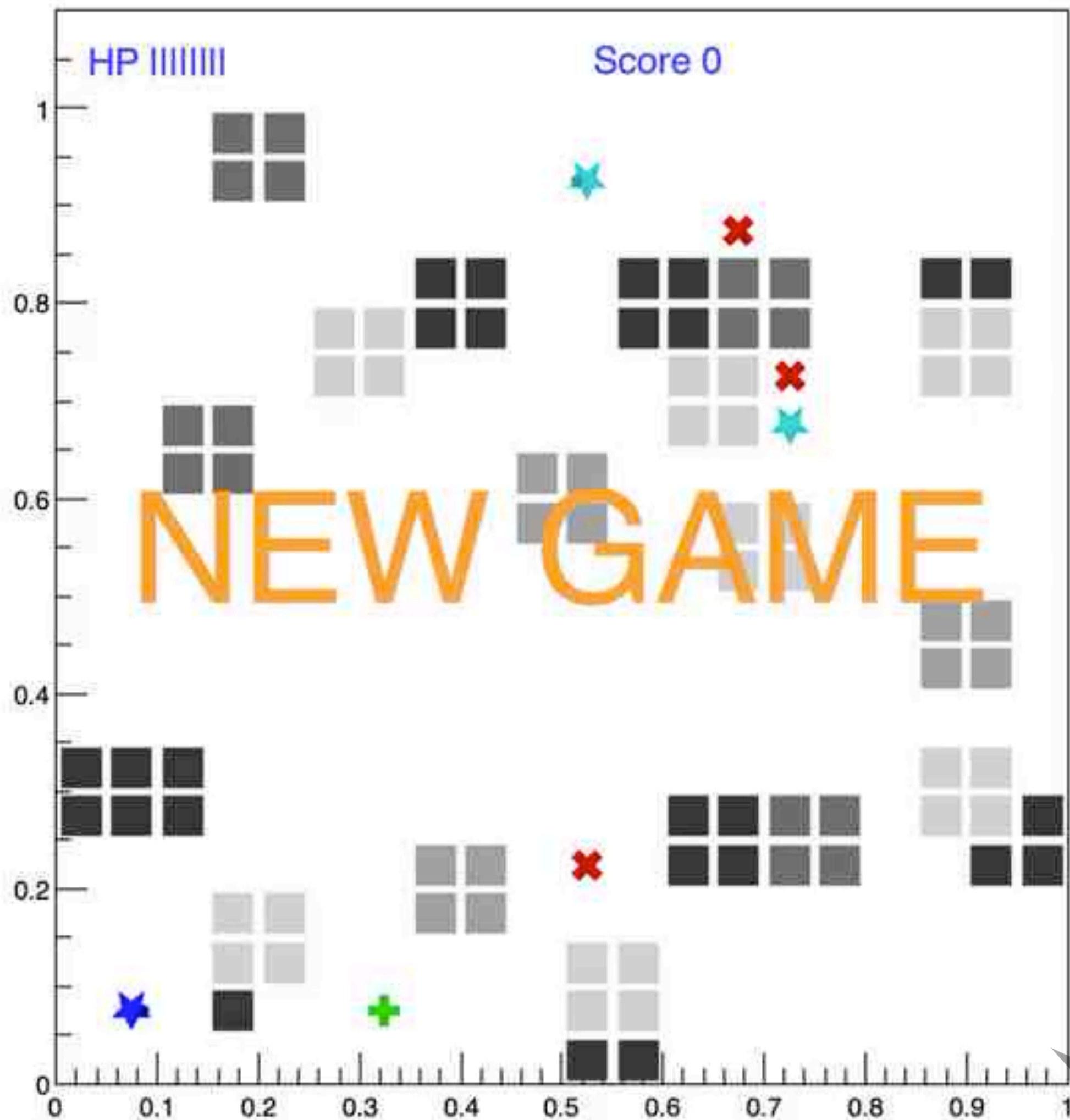


疫情結束後回到正常教室，
效果其實沒有太大差別！

❖ 線上期末海報活動：

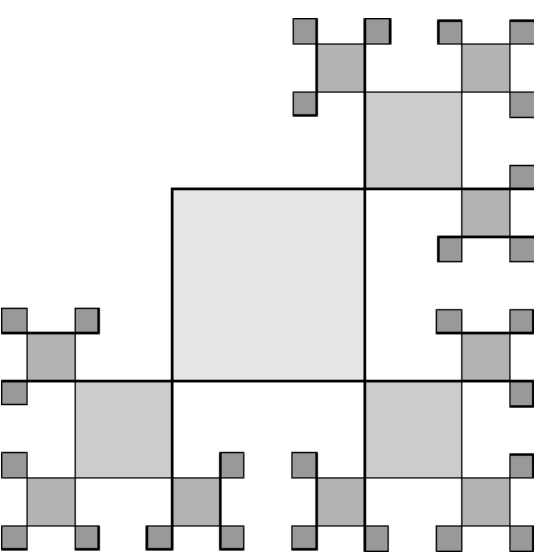


期末遊戲競賽

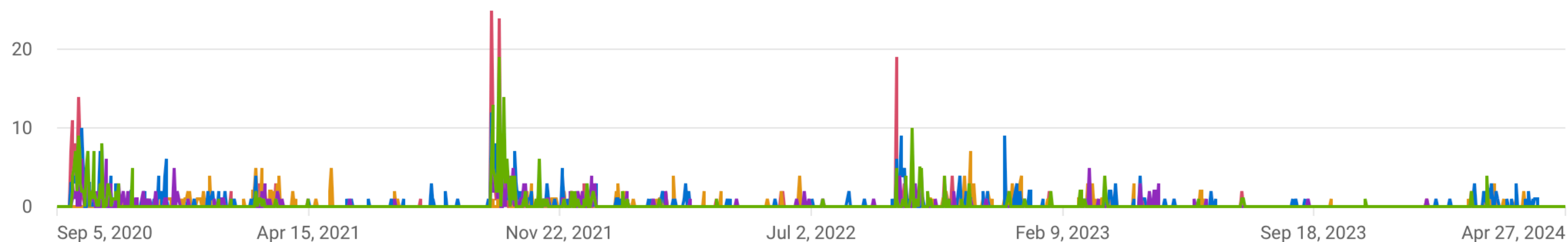


- ❖ 提供一類似古典電玩遊戲，但不是由人類操縱；**得要用參與者提供的程式判斷**。
- ❖ 學生會收到一個很弱的樣板作為起點，盡力改進演算法獲得最高分。
- ❖ 提供C++/Python介面，可獲得盤面所有資訊以進行判斷、回傳指令。
- ❖ 由TA統一測試，分數上打敗一半以上的參與者可獲「**獎盃**」。得分最高的會在學生前進行Demo。
- ❖ 第二學期會增加**機器學習的競賽**。

第一學期使用、類似「坦克大決戰」的遊戲



三年點擊統計資料

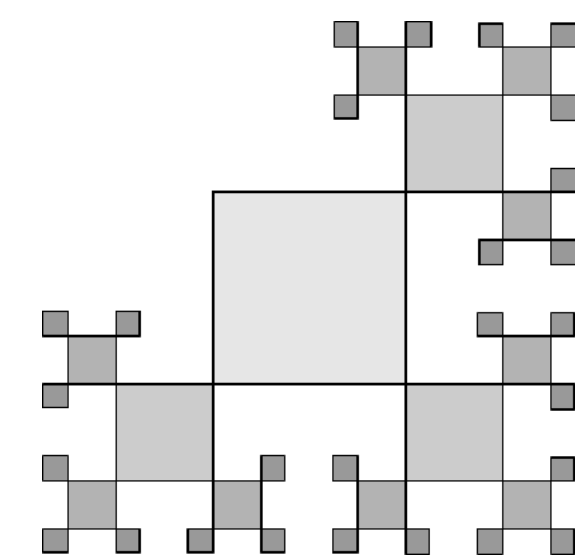


Content	Views ↓	Watch time (hours)	Subscribers ▲	Impressions ▲	Impressions click-through rate ▲
<input type="checkbox"/> Total	11,535	1,606.2	28	31,494	2.2%
<input type="checkbox"/> 47:01 Module C1: Into the C/C...	494 4.3%	64.0 4.0%	0 0.0%	864	8.2%
<input type="checkbox"/> 32:32 Module B4: Basic Unix ...	337 2.9%	59.5 3.7%	0 0.0%	426	3.3%
<input type="checkbox"/> 31:33 Module N1: Errors in Co...	278 2.4%	41.3 2.6%	0 0.0%	397	2.3%
<input type="checkbox"/> 29:08 Module P1: Introductio...	274 2.4%	34.6 2.2%	0 0.0%	481	2.3%
<input type="checkbox"/> 32:39 Module B2: Conceptual ...	266 2.3%	37.7 2.4%	1 3.6%	425	0.5%
<input type="checkbox"/> 36:30 Module N3: Numerical I...	263 2.3%	42.9 2.7%	0 0.0%	406	2.5%
<input type="checkbox"/> 28:26 Module B3: Tips for Pro...	255 2.2%	38.2 2.4%	1 3.6%	420	0.2%
<input type="checkbox"/> 21:02 Module N2: Numerical ...	246 2.1%	26.9 1.7%	1 3.6%	402	2.7%
<input type="checkbox"/> 32:03 Module C2: C/C++ Basi...	230 2.0%	35.8 2.2%	0 0.0%	867	2.8%
<input type="checkbox"/> 31:56 Module B5: Introduction...	221 1.9%	29.4 1.8%	1 3.6%	138	1.5%

❖ 根據Youtube點擊大概可以推估每個模組的受歡迎程度。

❖ 平均修課人數約50人

- ❖ 大致排序：
- 數值分析 >
 - 基礎工具(Unix, git) >
 - 程式語言 >
 - 機器學習專題 >
 - 計算物理專題



學生反饋

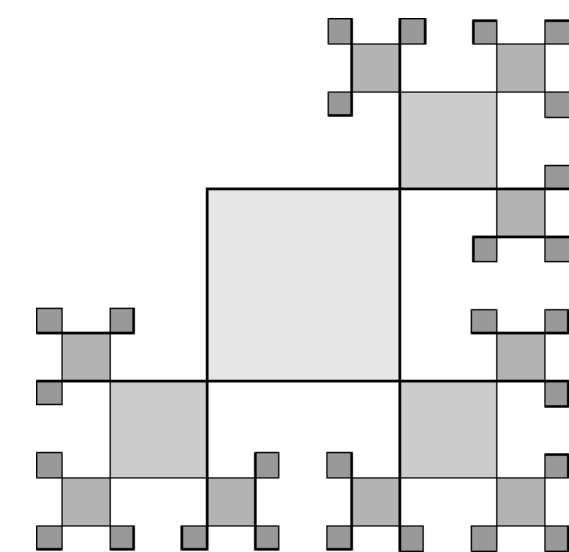
基本上偏正面，高自由度 + 遊戲方式是受歡迎的

Pro

- ❖ 很喜歡自己安排練習時間的做法。
- ❖ 很喜歡這種全線上的模式!可以有效運用自己的時間和選自己喜歡的學!
- ❖ 類似打遊戲破關的學習方式讓我很有學習動機，對於對程式沒有強烈興趣的我來說，若是以傳統的教學方式授課，我可能早就放棄了。
- ❖ 課程很有趣，還有遊戲式的競賽很有挑戰性
- ❖ 老師的網站真的做得很好，評分方式也非常有趣，讓我很有動力學習!另外，影片也講解得很清楚，覺得學到很多東西!
- ❖ 教授教學方法新穎且符合學習上的實際需要，沒有傳統程式課做無意義的練習的弊病。
- ❖ 這樣的上課方法超讚，讓學生選擇自己要學到什麼

Con

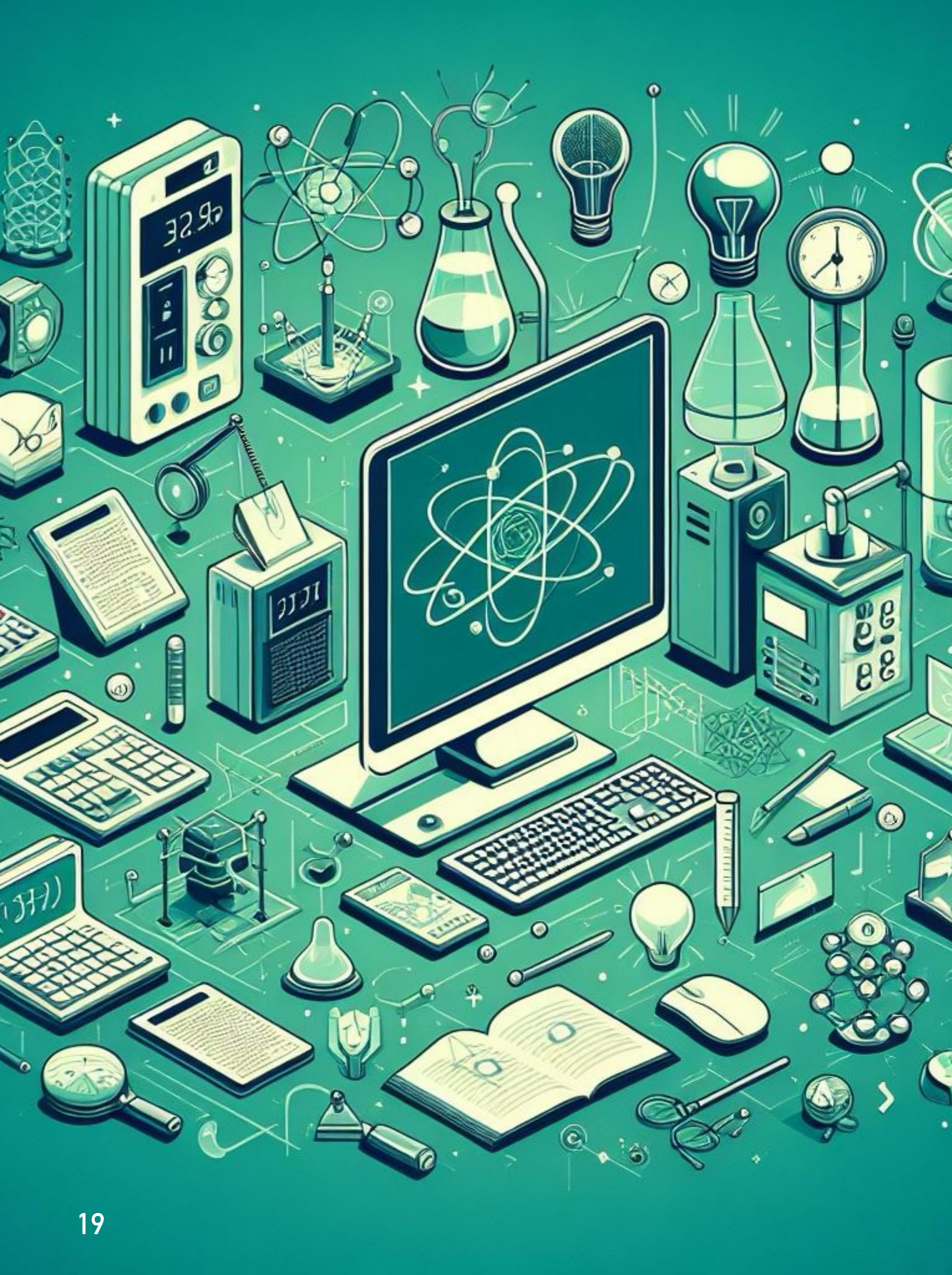
- ❖ 講課的內容較理論，可能要有更多實作的內容，否則對程式新手而言做EXERCISE很常會有不知所措的感覺。
- ❖ 對於不熟悉的程式最難的是找到如何啟動及如何繳交答案，希望能多加說明
- ❖ 在介紹程式語言邏輯的時候缺乏邏輯，導致像是在逐條背誦程式語言的各功能，並未強調背後的資料結構，並不適合做為理解程式語言的教學模式。



個人心得

- ❖ **備課非常困難**(尤其是準備習題、網頁伺服器架設與測試)，因為要能夠自動批改，要花很多時間debug。好處是一旦準備好就可以重複使用、比較輕鬆。TA非常重要，需要重頭到尾跟課；非常感謝我的TA幫忙很大！
- ❖ 選課學生分佈非常廣，大一到博班二年級都有，甚至還有一個中文系的來上課。到了第三年修課人數大減(也許是想修的都修過了)，**需要重新鼓催學生修課、改變物理系學生討厭程式設計心結**。
- ❖ 然而**學生的參與度相當薄弱**。多半的學生只專注做自己的東西，不管其他人、努力鼓催無用。報告時多半只有我和TA會提問。
- ❖ 即使在第三年納入「獎牌」的設計，吸引學生互動，最後會參與的學生還是不多(但是比沒有這個設計好多了)。





總結

- ❖ 為了強化大學部學生電腦計算相關能力，並對應COVID疫情，開設了一個全年的「計算物理導論」課程。
- ❖ 可以完全線上上課、依照翻轉教學設計。
- ❖ 學生有高自由度，可選擇部分材料消化。
- ❖ 所有預錄材料整合至單一網頁，所有的作業都可以自動批改、整合成績。
- ❖ 修課學生反應正面，然而在開滿三年後修課人數減少、仍須努力鼓勵學生修課、學習電腦計算相關能力。

