

壹、類神經網路

高一上，在閒暇之餘，去圖書館借了「[商用微積分](#)」以及「[初等線性代數](#)」，對[線性代數](#)以及[微積分](#)有了基礎認識，奠定學習類神經網路的[數學基礎](#)；又在朋友的推薦下學習 [Tensorflow](#) 套件，並應用在「[午餐系統及分析](#)」這項科展。

高三下，在個人申請結束後，我打算以 [Q-Learning](#) 建置一對[神經網路](#)，讓網路以 [UNO 遊戲相互切磋](#)，並做成一篇研究論文，投稿[旺宏科學獎](#)。

貳、專案開發

[小學](#)時，我特別喜歡「洛克人」這款遊戲，於是[我做了一個 Unity Game](#)，仿造「洛克人」這種 2D 卷軸遊戲。製作遊戲時，我學會了 [CSharp](#) 以及 [Javascript](#) 這兩種程式語言，還有如何找到所需的資料，這些技能[對未來開發專案深有幫助](#)。

[高中](#)時，在學校點餐必須劃記紙本點餐單，手動繳款，[有繁瑣又不穩定的人為操作](#)，若是能[以電腦取代人工](#)，豈不妙哉？於是，我製作了一套系統，協助班上點餐。

在[白翔云](#)同學的加入下，原先[架構不敷使用](#)，需要一套更完善、更健全的架構，於是完全捨棄舊架構，改成完整的 [3-Tier](#)(Frontend/Backend/Database)[架構](#)，使用 [Github](#) 進行版本控制，並推廣[給全校使用](#)。

這套系統就是「[點餐系統](#)」，製作這套系統，不只讓我學到怎麼[團隊合作](#)，還讓我學到如何[設計架構](#)，找出[效能瓶頸](#)，[需求釐清](#)，[系統分析](#)，[安全性評估](#)等等實用的技能。

參、算法競賽

開始接觸算法競賽是在高一，其中，我最喜歡的領域是資料結構，長久以來，我都一直秉持著一個信念。

「能夠以資料結構解決的事情，都不該用其他方式解決」

資料結構可以加速演算，降低時間成本；資料結構還能更有效的使用記憶體，降低伺服器負擔；資料結構更能夠降低解題難度，使問題迎刃而解。

熱愛資料結構的我，也有許多名言。

「怎麼求 LCA？當然是 Link-Cut Tree 啊」

「怎麼實作 DSU？要嘛 Treap 要嘛 Link-Cut Tree 吧」

「怎麼排序數字？丟進去 Fibonacci Heap 之後再拿出來就好了」

撰寫題解，協助他人解開問題，這也是算法競賽的樂趣之一，以下是我在 Zerojudge 上撰寫的題解。

題號	連結
c522	https://zerojudge.tw/ShowThread?postid=19466&reply=0
a121	https://zerojudge.tw/ShowThread?postid=18866&reply=0
a007	https://zerojudge.tw/ShowThread?postid=20495&reply=0

肆、資訊安全

國中時，看到 Orange Tsai 在 HITCON 的演講稿，學會了 SQL Injection，這是我第一個學會的攻擊手法；爾後，在好奇心的驅使下，我也學會了 XSS 等手法。

專案開發與資訊安全是相輔相成的一對兄弟，製作「點餐系統」讓我學到大量的資安知識，包括雜湊、非對稱加密以及第三方憑證，這些知識幫助我在資安競賽中走得更遠，也幫助我在專案開發中更得心應手。

伍、其他

一、經濟學

人們為什麼選擇去搶購口罩，就算要排一個小時？俄羅斯為什麼決定增產石油？義大遊樂世界為什麼決定降價到十元？這些都是經濟學研究的主题。

對這些議題好奇的我，參加了經濟學奧林匹亞，在準備的過程中，讓我學到了 Nash equilibrium、Pareto optimal 以及 Income inequality，也讓我理解這些現象背後的原因。

二、語言學

在準備語言學奧林匹亞時，我發現學習一門語言，先學習「聽、說」，再學習「讀、寫」，才是符合大腦運作的學習方式；我也發現世界上有很多語言不是十進位制，有十二進位制、六十進位制的語言；我還發現很多語言缺乏足夠的詞彙去表達意思，因此常常假借其他詞語來表達，如「有洞的地」等於井，「熱的豬」等於烤豬。

準備語言學奧林匹亞，讓我對語言學深感興趣，也讓我對「語言」本身有更深入的了解。

三、交通理論

在哪裡設置公車站能夠載到最多客人？為什麼在高速公路上踩緊急剎車會造成塞車？放行多少台車到高速公路上才能最大化運輸量？這些都是交通理論中的重要議題。

我很喜歡 Cities - Skylines 這款遊戲，這款遊戲的重點在於建設城市並規劃良好的交通動線；在家庭背景的薰陶下，我對交通理論漸漸感興趣，並從遊戲中學習交通理論，增加知識。