

程式設計大賽於 2004 年起第一次舉辦至今，已為 Openfind 年度重要的盛事及傳統之一。公司內部人員於比賽期間將打破原部門建置，重新編組，依據比賽題目內容，並在有限的時間及資源下，發揮最大的創意及團隊合作，努力達成目標。2009 年 Openfind 程式設計大賽花絮如下：

題目：Openfind 天使帝國. 比賽時間：24 小時

競賽規則：[Openfind 天使帝國比賽題目下載](#)

比賽結果名次公布：

- 第一名：Jamie, Berit
- 第二名：Felka, Yaoan
- 第三名：Zion, Henry

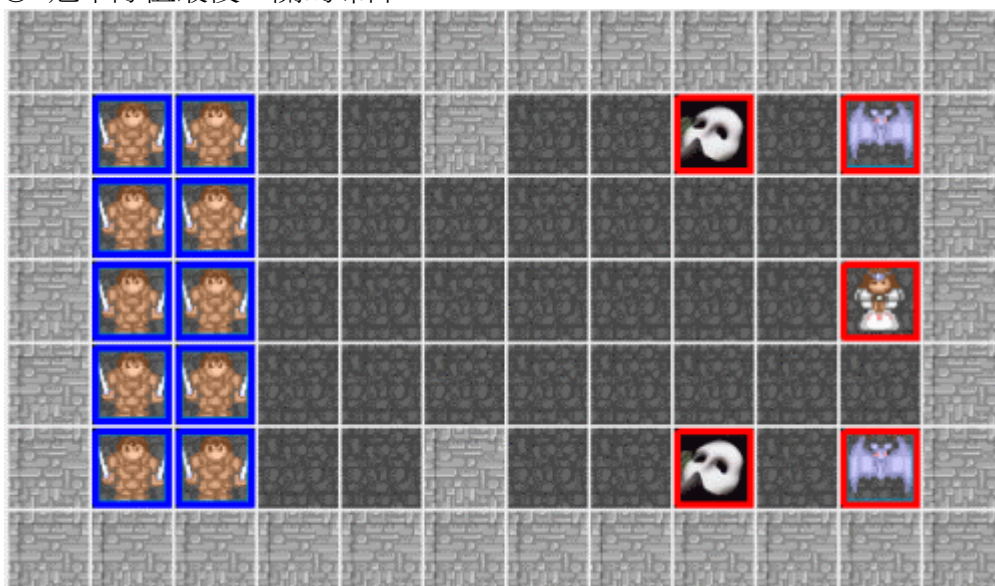
戰況剪影：

◎ 兵種列表

	Bat	Bat(G)	War1	Rogue	Mag	Priest
生命值	200	400	1000	420	190	550
攻擊	300	430	180	400	350	0
防禦	20	30	200	100	0	0
移動力	3	4	2	3	2	2
攻擊方式	近戰	近戰	近戰	近戰	法術(3)	治療(1)
移動方式	飛行	飛行	步行	步行	步行	步行
特殊能力	反擊可	反擊可	反擊可	反擊可	無視防禦	治療(200)
價格	200	400	200	200	400	200

	Golem	War2	Richard	Andric	Phantom	Ivan
生命值	1700	1400	2000	1800	1500	1100
攻擊	60	230	275	500	800	600
防禦	360	250	300	120	90	0
移動力	1	2	2	3	4	2
攻擊方式	近戰	近戰	近戰	近戰	近戰	法術(3)
移動方式	步行	步行	步行	步行	飛行	步行
特殊能力	反擊可	反擊可	反擊可	反擊可	反擊可	無視防禦
價格	400	400	1500	1500	1500	1500

◎ 冠軍隊伍最後一關的布陣



◎ 比賽結果

積分	第二名		第一名						第三名	
Round 1	9	5	4	2	6	1	8	3	7	
Round 2	9	4	6	1	7	3	5	2	8	
Round 3	9	5	8	2	7	1	6	4	3	
Round 4	8	5	3.5	1.5	1.5	3.5	7	9	6	
Round 5	3.5	9	8	7	5	6	1	3.5	2	
Round 6	5	7	9	3	3	8	1	6	3	
Round 7	1.5	3.5	7	1.5	5.5	5.5	8	3.5	9	
Round 8	4.5	1	9	4.5	8	2	7	4.5	4.5	
Round 9	8	4.5	9	2	4.5	4.5	4.5	1	7	
Round 10	8	4	9	4	4	4	4	4	4	
	65.5	48	72.5	28.5	51.5	38.5	51.5	40.5	53.5	

幕後花絮

Phantom	這次的比賽題日本來有兩個構想。方案 A 對我來說比較簡單，但是最後比賽過程將會十分冗長根本就是考驗體力。方案 B 就是這次的天使帝國，看起來比較有趣，也比較容易分出高下，只是我要寫出整個架構會很累。 最後經過跟 Andric 以及 Richard 的討論之後，他們兩位一看到題目馬上就選了方案 B，我也只好乖乖的認命回家寫程式。 為了記取去年的教訓，一個本來應該是策略比賽的東西居然最後變成穩定 coding 大賽，所以這次還沒做之前就決定要給參賽者絕大部分的 utility functions，讓大家可以好好的去思考策略。然後還特別的十場戰役的前三場公布出來，讓各組有時間好好的去測試，減低 bug 發生的機率。最後也證明這樣做是有效果的。只是好好一個比賽寫最多程式的居然是出題者，有時候會想這個比賽到底是在考參賽隊伍還是在考驗我？ 然後我還沒有錢可以領。Orz 決定好題目是戰棋類的天使帝國後，第一件要做的事情就是去找兵種的圖案，這時候我用了以前玩過的一個小遊戲「魔塔」，除了 Phantom 的圖形以外，最後所有的兵種都是從魔塔裡面抓下來的，有興趣的人可以玩玩這個遊戲。 第二件要解決的事情比較麻煩，就是關於兵種的平衡問題。戰棋類的遊戲我個人從國中開始接觸，之後有 Apple II 巫師神冠、春之石、PC-Engine 蓋亞的紋章、Sega 大戰略 I、大戰略 II、PC 單機遊戲的天使帝國，一直都是我個人最喜歡的遊戲類型，但是真正要考慮兵種平衡的時候卻發現好難好難，隨便把一個數值加 5 減 5 都會造成完全不同的後果，最後只好寫利用兩個 CPU AI 不斷的對打跟改良，盡量讓各個兵種都能有發揮的空間，甚至都有不可取代的特殊用途。也讓比賽的策略變化多端，各隊伍可以看到 CPU 每個關卡的陣式再去調整自己的布陣方式來取得最大的優勢。 整個系統架構、電腦 AI、關卡設計、以及最後的調整大概花了 60 小時的時間完成，我還故意把電腦 AI 寫得很笨，這樣才會有鑑別率。畢竟電腦可以用強力兵種、可以有遠超過玩家的預算購買部隊，如果 AI 再過份強大的話，搞到最後大家都是超低分會變成運氣大賽，就失去意義了。 最後由策略考慮比較周詳，系統也比較穩定的 Jamie + Berit 拿下冠軍。刻鋼板的 Felka + 想用暴力窮舉運算的 Yaoan 拿下亞軍。一大早就找出不錯策略的 Henry + Zion 拿下季軍。 而其實其他組表現也很不錯，最後可能是一兩行的 if-else 判斷移動位置說不定就可以變成前三名。另外有在網站上面公布了每一個關卡的解法(Phantom 寫的 user.dll)，有興趣的人可以去看看，我自己證明每一關都是有解的(只要你算對腳步)。也感謝大家辛苦的參加比賽。 ps. Phantom 真的很強，隨便派個兩隻就可以殺光大部分的隊伍。真不愧是我的月份。^_^ By Phantom, 2009/09/03 中元普渡
Core	這次的策略非常可惜沒有發生很好的作用，本來想要用打游擊戰的方式，每一個角色選擇最安全的地點躲藏，遇到敵人的時候再採用被動攻擊，沒想到地圖太小了，一下就被殺光，唯一值得慶幸的是蝙蝠殺法師和牧師，也就是唯一我們分數比較多的關卡...Orz，當敵方派出五隻法師，我方就派出五隻蝙蝠，躲在暗處一擊斃殺法師。

	另外，和 Jimmy 一組真的很開心 XDD~
Bart	以下是本次比賽完的感想： 這次比賽我和比賽資深的 RD-Jones 一組，剛開始心想，不會吧怎麼就我一個 QA，然後又兩個一組，有點對 Jones 不好意思，而且他整個很認真的寫，只能說我還是專心最好我 QA 的職責就是 Hero Support，專心的幫忙他測試，想些不一樣的戰略，隨然這次我們沒得名可是也是不錯的經驗，上次電梯比賽可能是穩定度比較重要，這次則是使用的戰略佔很大因素，可能我們就只剩一個人 Coding，所以很多戰略都不能實際寫出來測試他的成效，所以有一點劣勢的感覺，除了沒賺到錢，其他經驗還是不賴的，對以後在工作上遇到 Jones 也會知道該如何合作做事。 PS.居然出兩個 Phantom，可怕的是居然還有人打贏，大家真厲害
Felka	幕後花絮 基本上這次比賽的確是靠著精密的刻鋼板計算，當然不是靠一個 case by case 的寫程式。而是靠著報表來寫程式，Joyen 及 Zoey 大神教的。 一拿到主辦單位所提供的 execl 檔，就先分析角色與角色間的對打的損血情況，並且計算出每一個角色與角色間對打幾次後會 KO 角色，藉由兩大分析來提供攻擊順序，並且轉換成 C 語言的距陣，特別調整法師的攻擊順序，本來的攻擊 Priority 是法師->蝙蝠->Phantom->...，想不到有個賤人亂改 Phantom 的屬性，只好更改攻擊的 Priority 為法師->Phantom->蝙蝠->...。 事實證明還是球員兼裁判最厲害。 整個團隊的行為模式就是很簡單地找最近的敵人砍，並且攻擊模式依攻擊 priority 決定砍誰。 而這次為何前三項都是僥倖拿到第一，也是靠著單純地報表法，行為模式和攻擊模式一樣，而單純地更改成員來試試看得到的分數為多少，並且會把每次的更改寫入報表，而每次在測試修改的時候，都會注意填入報表的分數是否穩定成長，靠著一次一次的測試報表，分數一而再、再二而三的提升，這就是精密的刻鋼版法。 靠著報表提升自己的程式的穩定度是可靠的，在工程師的生涯裡面，使用報表提升穩定度是一個很好的途徑。這次能夠拿到第二純屬僥倖，感謝大家。
Henry	心得： 這次的題目就 coding 的角度來看比去年簡單多了，今年主要著重的部分是策略上的應用，包括一開始的佈陣、兵種的選擇、決定攻擊的目標，但是最後決勝的關鍵卻在於如何針對敵方每種 case 來使我方的策略最佳化，如果單以一種陣容走天下是沒辦法取得高分的，有效利用地形以及電腦 AI 的缺陷則是更進階的拿分方法。以上是看完大家

	DEMO 的感想，不過如果改在不知道敵方陣容的情況下對戰應該會更有趣吧 XD
Honesty	相隔兩年，終於有幸第二次參加程式設計比賽。 相較於第一次參加，其實有了很不同的感想。 以前總覺得，寫程式就是寫程式，比賽或者工作，都一樣，就是對著電腦敲鍵盤寫程式嘛。 但在這次的比賽裡，跟 Emma 討論分工完後，就完全沈浸在自己寫程式的狀態裡。 當然偶爾還是會跟 Emma 討論確認，或者四處晃晃打探消息、講講垃圾話，感受那種大家一起競賽的熱鬧氣氛。不過，最過癮的還是，可以有一整段大塊時間可以完全沈浸在寫程式的"flow"「心流」裡。工作上，其實偶爾還是會被各種不同問題、不同狀況給打斷，硬是得趕著處理不同的緊急狀況，但在程式比賽裡，就幾乎完全沒有其他需要分心的，就直接完全陷了進去，寫到後來，居然開始懷念起大三「deadline 海」時期，同學們一起窩在計算中心或者宿舍裡，熬夜把一堆原本是一星期的作業在一晚之內趕出來的「青春熱血快感」。 好像很遙遠的感覺了～ 人有點年紀之後，真的好像免不了就是會開始懷舊～ XD 另外，這次比賽比較不一樣的是有了 Phantom 好用的 utility functions，果然寫來極為順手，很容易就能實驗很多不會搞掛程式但又能展現不同策略的程式版本。但 Phantom 的機車還是完全展現在他豪華的角色能力及變態的佈陣，只能說，果然還是得夠機車的人才能在這種機車變態的遊戲裡勝出啊～～～ XD
Zion	這是我第二次參加程式設計比賽 很幸運又跟上次一樣是第三名 感覺這次的題目與上次參加的 Openfind 大海戰有些類似 不過這次又更加的複雜，兵種更多、地圖更小，可以躲的地方也不多 好在有前三關的題目、Sample Code 及許多好用的 Function 大大減少程式出錯的機會，又可針對前三關做最佳化 整個比賽最有趣的地方就是，決勝負的時候 可以看到各組在這一天來，所發展出來的策略甚至是陣式 每組都有他獨特的地方以及優缺點，從分數上就可以看出明顯的差異 果然沒有固定的策略才是最佳策略，沒有最好的只有最合適的 能在對戰時，自動選擇適合的策略，才是致勝的關鍵啊.....
Emma	程式大賽之前，其實很擔心，因為大家都是高手。 公佈題目的瞬間，心中 OS:不會吧! 寫遊戲!!! 對平常完全沒在玩遊戲的我來說，真是難倒了。 謝謝和我同一組的 honesty,給了我很多建議。 遊戲的選角與布陣真是一門大學問呢! (茶) 很特別的經驗，大家都辛苦了。
Jimmy	這次和 CORE 搭擋，過程相當輕鬆有趣，合作的相當愉快。

	第一次參加公司辦的程式比賽 一開始還滿緊張的，先把目標放在不要跑到一半當掉... 不過一開始寫基本功能就寫出一個找了一兩個小時找不到的 bug 還好後來有 Mike 幫忙找出 bug~ Mike 也都會一直把程式修正的比較模組化,function 化一點還有訂出共同使用的資料結構 讓我們後面的 coding 比較容易修改~ 像我一開始沒有訂好這些東西，所以中間就花了很多時間在把同樣的功能換不同的寫法 然後覺得兩個人一起寫程式很好玩 可以互相 debug，自己腦袋不清醒的時候還有人可以提醒你 邊寫邊講解給對方聽，或者在講解寫好的 code 時也是把自己的思路再整理一遍 尤其在夜深人靜腦袋不清醒的時候講講話好像會比較容易醒著(?) Mindi 還有我們都還在 implement 最基本功能的時候就聽到有人又找出了什麼系統的 bug 或者打贏了 15 個 richard 之類的消息就覺得越來越緊張 XD 後來覺得布陣跟部隊配置很重要所以就做了一些實驗 不過可能是自己一直只會用一些很極端的測資，像是 15 隻的每一種 type，或者自己想的到的情況有限最後配置的策略過於簡單，效果也不太好 冠軍那組幾乎只用同一種陣型就拿很高分，還打贏兩個 phantom 真是太厲害啦 還有最後一個小時和 Mike 想到要去追擊 priest 就開始寫 可惜到最後有 bug 沒有 de 出來 所以最後還是只能跟 priest 在關卡兩邊一直跳來跳去互不侵犯 很高興可以參加這次的比賽覺得很好玩 能和 Mike 一起合作把策略一步一步 implement 出來覺得很棒 還有最後大家一起看每組測資結果也超緊張刺激的 :p
Berit	一年一度的程式設計比賽又來到，按照往年慣例，今年的分組名單經過 Phantom 的黑箱作業之後揭曉了，感覺是相當有趣的組合。雖然 Phantom 堅稱這是神明的指示，不過我想在鬼月請示神明應該是只能得到好兄弟的回答。總之，有幸與家民同一組。為了不給自己太大的壓力，輕鬆參賽，我們將目標訂在前八強。(因為廣告說的好，先講求不傷身體再講究效果，其實參賽隊伍也才九隊...) 比賽開始，題目揭曉，今年的題目是老少咸宜、大小不欺的「Openfind 天使帝國」，是個以策略為主的題目，還是個能讓人在歡樂中比賽的遊戲。在分析過 CPU 攻擊行為，並且經過連番的討論之後，我們擬定了基本的作戰方針：採取守勢為主，配合兵旗佈陣與兵種相剋的論點，佈下死亡陷阱，集中攻擊；並適時易守為攻，主動出擊，徹底掃蕩！(這方針有夠囉嗦，一點都不「基本」...) 經過一夜未眠的努力與好兄弟的保佑下，我們總算是有點成果出來了。我的感想是：雖然家民和我長相沒有 Mike 老實，身高也不一樣高，但我們都懂得合作的重要性。這次能得到不錯的名次，最大的原因便是我們能夠充分溝通與發揮團隊精神所帶來的結果！ (首先我要感謝我的父母...泣...)
Shawn	進公司不到三個月遇上了公司的年度大事：程式設計大賽。本以為照著西洋象棋以及大海戰一樣，都使用簡單的策略就不會太糟糕。

	沒想到題目居然是完全不熟悉的 SLG. 在完全不能理解各職業特性用途的狀況之下，我用了每單位能力價格來選出看起來 C/P 值最高的職業，陣形也很單純的排排站，而移動的策略就各自為政，尋找能夠打掉敵人最多血的位子。 由於除了前三關外不清楚 Phantom 關卡會如何安排，也不太懂這種遊戲該如何佈陣，所以 test case 我都只有很單純的單一兵種，沒想到在這種狀況之下我的策略就已經打得很吃力了。 接下來我就就分析了不同職業之間的相剋關係，打算在一開始的佈陣使用。 可是由於很久沒有熬夜了，到了半夜頭腦開始不清楚，自己在寫甚麼自己都不知道，就寫出了一堆 Bug，到了早上還沒有解決，最後與 King 的程式整合在一起，本想說在某些狀況下用我的策略，沒想到真正上場的時候我的策略根本沒有出場的機會，也就是說到最後我等於是完全沒貢獻了，實在是太對不起 King 了。 不過這次比賽還蠻好玩的，也得到了不少的經驗，希望下次可以有機會能派上用場，(說不定在那之前得先多玩一些遊戲吧?)。
Jamie	上次比賽慘敗後，這一年可說是籠罩在失敗的陰影之下，一直掙扎著要不要參加下次比賽。 如果沒有好表現，心理上的難過可是遠大於身體上的疲累的。 但也不想錯過可以跟其他同事合作切磋的難得機會，也想給自己一個機會再表現看看。 可說是對程式設計比賽又愛又恨，隨著比賽日期的到來，不禁讓人感到「近賽情怯」。 這次比賽很巧的跟王排(王牌!?)(也就是 Berit)同隊。 還記得公佈組別前幾天我們一起吃飯時還在講說，會不會我們兩人一隊啊，我還說，如果我們同隊的話，我要用 C++ 喔。 沒想到我們真的就同隊了，實在是蠻幸運的，因為平時就很熟了，合作起來相信會蠻順暢才是。 公佈後，我們的目標就是要進前八強！ 比賽前一個禮拜，我們各自在家複習基本資料結構跟演算法，我也複習了一下 C++ STL，希望比賽時不要再花任何心思在打造基本的資料結構上，像是 string, vector, list, set, map, stack, queue, priority queue。 不過，比賽沒用上，基本的 array + struct 就夠用了，不需要太複雜的資料結構。 用 C++ 只是用了一些語法蜜糖，像是 bool 跟 reference (作 out 參數還蠻好用的:P)。(題外話，STL 高效又好用，為什麼沒人在用，嗚嗚) 告訴自己得失心別放太重，只希望有正常表現，兩人也能愉快合作就好。 當然還是要進前八強就是啦～ XD 比賽前一天我們特別早睡，希望能有最好的精神來比賽。 比賽開始前瞄了一下加油網站，哇勒，竟然沒有人下注我們～ 瞄的～看到這種情形，沒有人心理不會幹的，我跟王牌都超怒的。 原本打算平常心應戰，此刻我們也不得不化悲憤為力量卯起來拚了啦～ 決定不理會外界的紛紛擾擾，專注當下，面對最大的敵人 - 自己。 不過比賽途中，還要應付其他人的冷嘲熱諷，真是幹到不行啊 啊 啊～ 我們一開始先個別看投影片，看程式碼。 為了更了解程式，我花了點時間加 debug 機制，印出一些想看的變數，也方便後續 debug 之用。

有了基本的了解後，接著花不少時間在討論陣形及策略，我們把地圖方格印在紙上(印好幾個空地圖)，在上面畫圖模擬討論(紙上談兵？)。除了討論出一般陣形外(對於 CPU 混合兵種的預設陣形)，也討論到敵方全為戰士系時的最佳陣形(肉盾 + 魔法師 + 牧師)。

一般陣形圖示如下 (# 為牆壁)：

W M #

W

W

W

W M #

原則上就是魔法師躲在牆後作遠程攻擊，周圍各派兩個戰士保護，中間再補個戰士，故意設個凹口，敵人衝進凹口就可以五打一。

後來覺得透過紙上談兵來討論陣形還蠻有效果的，畫出來不僅方便溝通，便於沙盤演練，優缺點及注意事項也可以在旁邊標註。

有了初步的共識後，就準備著手來實作一般陣形。特殊陣形等一般陣形出來再說。這時尷尬的情形發生了...

陣形的實作，一個人就可以了，另一個隊友怎麼辦呢？

我很想親自動手，於是我提議 pair programming。

然而王牌這時作了一個關鍵的決定...

他自告奮勇要實作「決定最佳攻擊對象」：

====

CalAttackPriority():

從 g_map 中，在我方部隊不移動的狀況下，算出最佳攻擊對象

計算的大概原則是，被打越多血越佳。若會被打掛的話，就看誰價格比較高。

(其實會算出所有攻擊對象的分數，再回傳排序後的結果。caller 就取第一個即為最佳)

====

等於是從上而下鋪陣形，定出整體架構，他從下而上打起決策的基礎。

如此，兩個人的貢獻不會重複，不會浪費寶貴的時間，能確實發揮團隊的戰力。

能跑的第一版出來後效果不是很好，可說是...被打得落花流水。

不過我們不在意，因為實作還不完整，

像是沒有在移動部隊到陣形戰鬥位置時順便作第一擊，以及永遠不會衝出去殺敵，還有 CalAttackPriority() 尚未處理敵人都會被打掛時該選誰的決定。

接著我就補強移動到戰鬥位置順便作攻擊，事先修改 g_map (假設各部隊就戰鬥位置)再利用 CalAttackPriority()，就可以決定每個部隊第一擊要先打誰了。

在此同時，CalAttackPriority() 功能也完整了，可說是在大部分狀況下都能作出夠好的選擇的 heuristic。而且沒 Bug，讓我們能放心地仰賴它。實在很佩服王牌。

為了實作衝出去殺敵，問題就來了，

一個部隊有那麼多步可以走，每一步都可能殺到不同敵人，

該走哪一步殺哪個敵人呢？

有了 CalAttackPriority()，這問題就變得很簡單。

針對一個部隊，對每個他能走的步，試走，再呼叫 CalAttackPriority() 得到該步最佳攻擊對象，接著再比較每個步的最佳攻擊對象，就能得到所有步中最佳攻擊對象，以及該步為何。

====> TryAllMoveAttack()

在此同時，我的最佳拍檔當然沒有閒著，
除了完成判斷敵軍陣容的邏輯外，
他正在實作另一個關鍵功能：「決定最佳補血對象」
==> CalHealPriority()

我真的不得不說，王牌真的是太可靠了！在我昏迷的時候，能頭腦清楚地主動找有用的功能來作，並產生高品質的結果。真的是個很優秀的隊友。

但是在這個時候，其實我們的進度相當落後，因為途中一直聽到其他組已經順利幹掉前三關，在 tune 其他未知的 case 了，而且甚至能幹掉 15 個 Richard!

但是...

不管!!!

不管外界的紛擾，專注於當下，我們的敵人是自己啊!!! 不管!!!

雖然這麼說，可是心理還是非常不安啊 啊 啊～

畢竟我們前兩關還是打不好，

只有第三關打得好 (事後證明，比賽時該關卡拿到第二名，是真的不錯)。

實作出 TryAllMoveAttack() 後，我還來不及調整上去 就昏迷了...

也不知道昏睡了多久，在遠處漸漸傳來一些聲音，是別組討論的聲音，

內容似乎是調了好幾個陣形，可以打多少分，效果很好...

我就驚醒了，整個精神都回來了！

已經快凌晨七點了，第一關第二關還沒過，天啊。

兩人作了那麼久努力，好歹也要打進前八強啊。

精神抖擻，趕緊測試 TryAllMoveAttack()，稍微調整原本的陣形，只是預設都往前衝。

結果發現，效果不錯！一二關都能順利過關，分數也不難看。

此時士氣大振～

下一步相當重要，經過討論，我們認為沒存在一體適用的陣形，

一定要針對 CPU 不同的兵種變化作出相剋的陣形策略。

因此接下來決定針對 CPU 各種兵種變化，擬出對應陣形策略。

之前已經想過全戰士系的 case，接下來要盡可能補齊各種變化。

王牌早就已經想過各種變化了，只差還沒擬出策略。

(全戰士, 全魔法師, 全蝙蝠, 戰士 + 魔法師, 戰士 + 蝙蝠, 魔法師 + 蝙蝠)

接下來，我就先實作針對全戰士系的陣形 (肉盾 + 魔法師 + 牧師)。

王牌就開始設計 CPU 兵種變化的 test case，並著手擬策略。

[肉盾 + 魔法師 + 牧師]的陣形實作出來後，可以成功擋住兩個 Andric 搭配其他戰士，確定對上全戰士系為必勝陣形。

士氣又大振～

此時，足智多謀的王牌已經擬定好策略了，真是太棒了！

對上全蝙蝠，用十個盜賊就可以輕鬆應付，而且第一場先不動，等敵軍靠過來後，再出去攻擊，效果更好。預設就是直衝直打就是了。

有了先前完成的 TryAllMoveAttack()，這策略實作起來超快 (其他策略也是)，有倒吃甘蔗的 fu。

我們用兩個 Phantom 來測，有一個 case 能成功過關，另一個 case Phantom 放不同位置就只能殺一隻了，無論如何，算是夠好了。

(士氣持續上升中)

	總之，我們在兩個小時左右就實作了三個策略，總共有五個策略來因應。 最後，我們發現，耶？本來不是要在一般陣形破掉了後，讓部隊要衝出去打嗎？一直實作新的陣形策略，都忘了基本陣形需要補強的地方了。 加了判斷陣形破掉的邏輯以及部隊往前衝的行為後，這個陣形可說是完整了。 再次測試後，確定所有 test case 都沒有問題，就決定最終版本了。 此時已經是 11:30 了，距比賽開始只剩 30 分鐘，交卷了。 比賽開打，前四關打完，除了第三關打得好外，其他都打得不是很好。 不過我們像是倒吃甘蔗，後面六關都打得很好。 很幸運猜到兩個 case (全戰士 跟 有兩個 Phantom 的全蝙蝠)，也都順利都拿到高分。 其他關卡由一般陣形出戰也有亮眼的表現。 最後能拿到好名次，真的很開心。 我們程式能有不錯的表現大多要歸功於身強體壯，主動積極，細心穩定，足智多謀，講話北七的好隊友王牌。 很幸運可以跟他同隊，我們溝通相當順暢，所有的決策都經過充分討論，雖然一開始花了不少時間，但相信對我們產出的品質有相當的助益。 合作過程也相當於愉快，充分發揮了團隊戰力。 此外，他體力超好的，我好像沒看到他在睡，頭腦一直很清楚，作出許多明智的設計決策。 也能主動找出需要加強的地方補強。又能有高品質的程式產出。 在熬夜比賽時還能保持這樣的穩定度，真的很不簡單。厲害。 除了把基礎的決策打好外，又能針對 CPU 各種兵種變化擬定有效的應對策略，穩定的底層跟上層策略的結合，讓我們程式擁有相當好的強度。 果然是名副其實的王牌。 真的很感謝他的努力，也很高興能跟他合作。 最後，要謝謝 Phantom 設計出這麼好玩的題目，大海戰已經夠有趣了，這次卻有過之而無不及，而且比大海戰沒爭議。 相信下次的題目會更有趣～
Mike	有鑑於上一次的到早上時已經把 phantom 看成言承旭的神智不清，這一次刻意在拿到題目後就開始找 mindy 討論。 這次的題目較上一回不一樣的地方是，程式都寫好了，剩下的就是兜策略。 老實說，這個題目對沒在打遊戲的乖寶寶來說，真的是未打先輸一半。 但沒關係，我們有 openfind 最新鮮的肝...不是，是最強的新人 mindy，相信一定可以有所作為。 一開始我們並沒有在組隊上下功夫，而是先把基本的策略先刻出來。 這部分 mindy 真是太優秀了，下午我不過出去準備福委會的餐點，回來 mindy 就把程式寫好了。

然後接下來隨著月亮越高，我的神智也越來越不清。
所以之後的組隊方式、跟加入治療系的隊形，都由 **mind**i 一手包辦，
我只能負責提供一些有 **bug** 的程式而已。

這裡很不好意思的是讓 **mind**i 也住在公司一個晚上，
我想這部分 **phantom** 應該可以補貼一些精神賠償費，
畢竟 **phantom** 月跟 **phantom** 在一起，還是會被嚇到的。

以上，兩個人一起生一個程式我覺得比較有合作的感覺，
這樣可以激盪腦力，以及偷學一下別人思考和寫程式的方式。
我相信這應該是鬼神 (**phantom**) 的智慧所要我們學習的地方，
而不是單純地只想出題目，然後把大家電得吱吱叫而已。