hw4\_a1045516 心得

系級 : 資工系108級

學號 : A1045516

姓名 : 蔡湘俊

這一次的作業可以說是十分的有趣，而且它也是Link List的活用，藉由Link List可以從尾端循環回開頭的特性，讓我們能夠以圓環的形式下去找尋要被殺掉的人，假如今天是用陣列寫的話就會變得很麻煩，因為陣列是沒有這種可循環的性質，而且這份作業可以說是十分的有挑戰性，原因在於說今天這份作業是有去限制程式的執行時間，而一開始在寫的時候對這個時間的限制是不以為然的，因為自己想說電腦的運行速度是十分的快速的，應該整個程式跑完是不會超過15秒才對，又加上k值是十分的小，所以就更加相信這樣的想法，可是當我開始運行k從1到13的時候，我發覺到隨著k值的增加，雖然不是全部的n以遞增的形式增加，可是那也只是少部分，尤其當k大於等於10的時候，所得到的n可以說是以倍數在做成長，使得我整個程式跑得十分的久，更甚至跑了好幾個小時結果也都還沒有出來，而這時我才發覺到說用暴力解的方法肯定會超過題目所限定的時間。

在發覺到了這一點之後，我就改換用遞迴的方法下去寫，可是在用遞迴的方法的時候可以說是發生了一堆狀況，整體看起來的感覺像是遞迴一直跑可是已經不知道跑到哪裡了，導致我不斷的發生無窮迴圈的情況，之後確實有改成，可是跑出來的時間仍然是大於15秒的限制，而且k值都還沒有超過9，之後想到說遞迴雖然很好寫，可是其執行的時間是會花蠻久的，而如果去用迴圈的方法下去寫的話，就會比遞迴再快上許多，所以我之後就改成了迴圈的寫法。

可是在改成迴圈的寫法的時候，確實是有跑出結果出來，而且也都在15秒內，可是顯示出來的結果都是錯誤的，當下可以說是整個毫無頭緒，明明該判斷的都有去做判斷了，可是還是會發生取到空值或者是移動到已經被殺掉的人的位置，而之後再深入的去思考的時候，我發覺到我自己把整個判斷條件想得過於複雜，就整個判斷而言，只要去判斷說，假如我移動到的下一個位置是空的話，就將目前的位置指向到開頭的位置，這樣子就形成了一個迴路，並且也可以避免取出來的資料會是空的，那假如我移動到的下一個位置不是空的，可是這個位置已經被殺掉了，那麼這時只要去用while迴圈下去跑，來去找尋還沒有被殺掉的位置就行了，可是在這尋找的過程中，一定有可能會發生已經跑到尾端了可是卻還沒有找到存活的位置，這時就必須在while迴圈裡頭多補上如果跑到尾端的話就回到鏈結的開頭，這樣我就可以去確保到說我移動到的下一個位置必定不會是空值並且這個位置一定還是存活的，而且這樣子下去寫還可以去避免說迴圈的次數因為空的位置或者是被殺掉的位置這兩個因素的影響，導致迴圈的次數都會少了一些，使得最終的顯示結果都會是錯誤的。

在這次的這份作業當中讓我體認到，不能總是去想著說現今電腦的運行速度都十分的快速，就只要把全部的工作都丟給電腦然後等一下子就能夠得到我們想要的結果，像這次這份作業之所以會要去限制時間的原因就是要避免我們去用暴力解的方法下去將題目給解完，因為這樣的方法不但是最沒有效率的，而且其所需花費的時間都有可能高達到數十倍之多，而用了優化解之後確實將整體的運行時間給大幅度的降低下來，更甚可以達到瞬間跑出執行結果的程度，而在現今的許多軟體確實是需要優化解的存在，像一些醫療軟體或者是監控軟體等等，這些軟體都必須跟時間來去做賽跑的動作，任何的一分一秒都不能平白無故的浪費掉，因為往往就在一秒之差更甚至是到毫秒之差，都將會導致不可想像的意外發生，所以時間對於這些軟體而言就變得極其重要，不過其實現在儘管電腦的運行速度越來越快，可是對於使用者而言在使用軟體的過程也是希望越快速越好，理由很簡單，在這科技不斷進步的發展下，我們對於任何的事物已經不再是追求有就好了，更多的是注重到該事物的本質以及呈現出來的結果是否是符合我們所期待的，就如同一旦我們使用過好的東西，我們並不會希望它一直維持目前的狀態或者是更糟，我們更多的是希望它能夠為我們帶來更多的變化、驚喜、效能與實用性，就如同大家常在講的好就還要更好，而這也是做為資工系學生的我所必須去學習與體會的地方，因為在未來的研發只能比之前的更好不能比之前的還差，否則所研發出來的東西是不會有人去買單的。