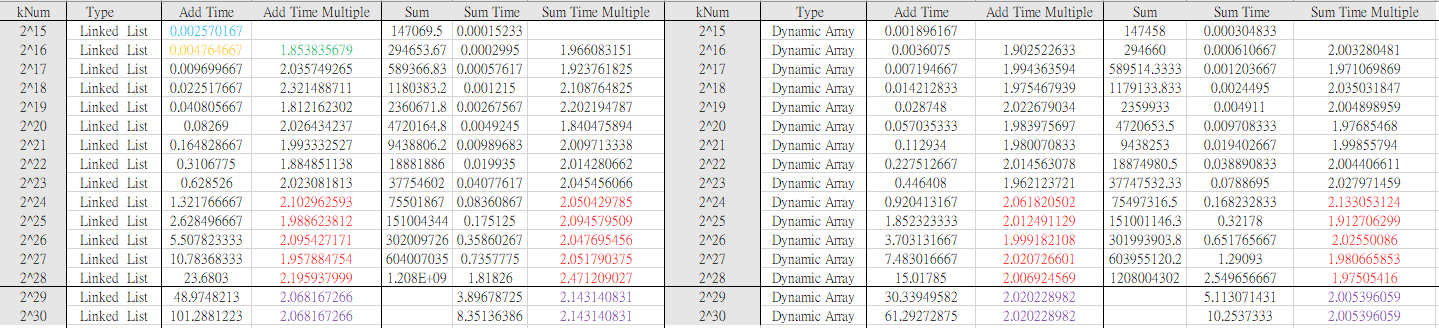
**作業一**

**109208064 資科三 俞懿**

**作業目標：比較dynamic array與linked list。**

1. **(60%) 兩個折線圖（每個圖包含2條折線，分別代表dynamic array與linked list所需時間）。  
   折線圖的x 軸為，y 軸為時間。請解釋你的實驗結果。**

下圖為實驗結果的資料與推測 ，因電腦只可測到的資料，的資料是我的保守預測。



預測的資料的方式是，計算前五項資料數值增加率的平均，如圖Add Time Multiple / Sum Time Multiple欄位，紫色數值是紅色數值的平均。

Multiple(綠色數值)的算法是當列數值(黃色數值)除以前一列數值(藍色數值)所得的倍數。

因此的數值是的數值乘以平均所得Multiple的結果，而的數值是的數值乘以平均所得Multiple的結果。

折線圖的y 軸為時間，x 軸的最上排為，最下面兩排為多次實驗結果的平均數值。

由此兩圖可知在資料數4194304(2^22)之前的dynamic array 和 linked list 的差異不大，新增資料和計算總和所需的時間沒有差很多。

但在資料量更大的情況下，dynamic array 的所需的時間逐漸比 linked list多，因此可推得在新增大量資料的情況下 linked list 比 dynamic array 還來的有效率。

另外由上圖可知，兩者平均而言，特別是資料量很大的時候，加總資料所需的時間比新增資料所需的時間還來的短。

**2. (10%) dynamic array與linked list的程式碼來源，並解釋這些程式碼確實實作dynamic array與linked list。**

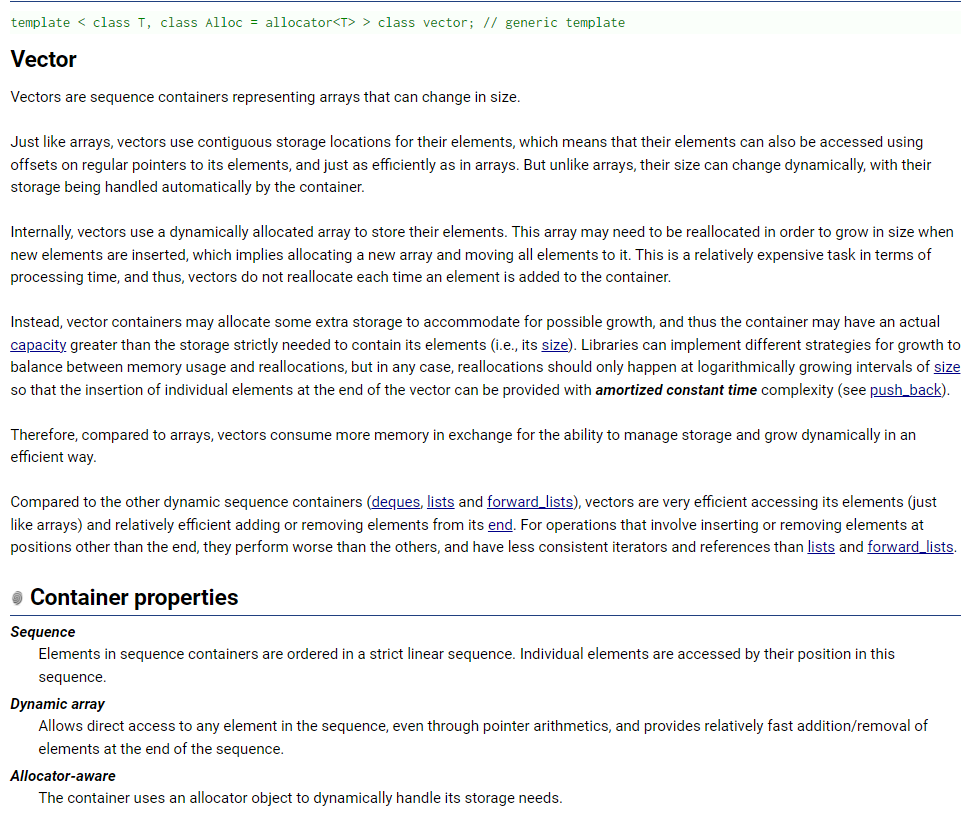
Dynamic Array 的程式碼:

來源1 : <https://linuxhint.com/summing-vector-elements-cpp/>

來源2 : <https://shengyu7697.github.io/std-vector/>

vector是C++標準程式庫裡基本的容器。

Dynamic Array具有以下性質，當插入新元素時，若現有陣列不夠存放，則重新創建陣列，建立一個更大的陣列，並將所有元素移至該數組。而vector也同樣具有這項功能，因此可作為dynamic array。

以下為<https://cplusplus.com/reference/vector/vector/>的介紹。

Linked List 的程式碼:

來源 : [**https://www.geeksforgeeks.org/sum-of-the-nodes-of-a-singly-linked-list/**](https://www.geeksforgeeks.org/sum-of-the-nodes-of-a-singly-linked-list/)

來源網站的程式碼結構具，每新增一筆資料時，list就會多new 一個Node，存資料，再用pointer與原本的list連接。此符合linked list的性質，因此確定為linked list。

**4. (10%) 實驗的程式碼。**

<https://github.com/yvonne90190/Data-Structure-HW/tree/main/hw1>

**5. (20%) 心得、疑問、與遇到的困難。**

這個作業包含了很多檔案，包括從網路上查的dynamic array, linked list的程式碼，還有main function，因此如果要通通全部在一個檔案裡的話就會很大個、不好整理，因此我想把它分成.h，.cpp等檔案，並且用makefile來執行，但因為以前CP1 , CP2的作業沒有這麼大，因此對於這樣的做法很不熟悉，又加上今年第一年寫c++，要怎麼處理class的.h、.cpp分配很不清楚，因此上網看了很多資料，我才知道原來stuct 要放在.cpp檔中，然後在.cpp檔中某class的函示前面補上class，然而當我把以上問題解決之後仍然有錯誤，後來問了學長才知道是template的問題，把.h檔的include改了一下馬上可以，實在太神奇了 !

另外我還問了學長怎麼寫makefile，雖然在CP2的最後一堂課有教，但我那時候只在趕期末專案，完全沒有認真複習，在他的指點之下，終於寫出一個比較好看又方便的makefile了。在執行程式時，因為有makefile，效率感覺大大提升，不再需要寫很長的指令了 !

總體而言，我一開始感覺很害怕，感覺這作業聽起來超難，但真的做之後感覺就沒那麼難了，而且還很好玩 ! 也很謝謝學長的提點，以後我也要跟他一樣學會很多東西，然後在別人有困難時教別人，助人為快樂之本 !