

以下是使用工作站跑出之部分結果 (case 1 2 3 4 7 8)：

$\lfloor 21$	$\lfloor 3330$	$\lfloor 20248$	$\lfloor p$	$\lfloor 10442$	$\lfloor 69360$
2 4 1	9 35 -63	42 20 90		26 21 21	263 198 -94
2 1 1	21 24 55	61 83 41		8 27 -7	354 244 17
8 1 1	49 22 -18	2 95 -41		1 23 56	254 462 -83
8 4 1	33 3 -98	51 47 53		23 18 29	248 400 -53
8 9 1	30 33 -67	12 2 -3		29 8 -55	311 410 -57
8 5 1	47 16 -94	28 78 52		12 3 -77	309 204 -58
8 6 1	35 10 -15	0 63 -54		5 11 -99	2 201 -88
9 7 1	3 27 46	27 55 -85		9 17 27	128 9 -19
9 5 1	24 19 23	64 95 22		23 10 37	32 386 -73
9 1 1	8 6 -25	57 97 -96		22 1 96	375 173 -30
9 6 1	41 18 5	31 12 -77		22 4 8	221 150 -90
9 3 1	7 3 46	72 53 -1		28 0 -73	157 10 -49
6 4 1	9 29 -75	48 76 14		16 13 9	487 474 -59
6 1 1	40 0 -68	2 49 -79		8 15 35	433 85 -90
6 7 1	40 7 19	45 6 -82		14 13 -6	92 18 -14
6 5 1	42 16 -26	54 18 -26		4 9 -7	150 441 -52
1 3 1	26 22 1	61 54 -47		22 12 -6	199 463 -40
4 7 1	17 12 40	79 58 -69		15 20 -35	45 235 -40
4 3 1	31 38 -34	41 39 -39		6 1 21	221 350 -43
5 7 1	13 21 27	21 32 -75		6 2 84	274 248 -23
5 3 1	22 14 5	15 20 -87		20 13 33	89 485 -40
	14 0 24	34 7 -29		19 28 61	282 194 -82
	4 27 64	68 92 -58		28 26 64	470 443 -30
	44 7 47	37 27 -21		24 8 -91	127 94 -54
		70 73 -20		12 25 -32	

資料結構及演算法 & 結果分析：

在這次的程式作業中，需要實做三個不同的問題，分別是 (1) 無向無權重圖、(2) 無向有權重圖，以及 (3) 有向有權重圖，其中有使用到的資料結構有包含了 vector、array、pair 等等。第一個問題主要的實作方式是使用 BFS 演算法搜尋並回傳要刪去的邊，由於是無權重，我們不需要考慮大小（一條邊總權重就加一），僅須確保其結果為無環即可。在第二項作業中，考慮到各邊權重大小不一，我使用 priority queue 資料結構來實作處理 MST 問題當中的 Prim's 演算法（原本有考慮使用 Kruskal 演算法，不過 disjoint set 的資料結構實做上較花時間）來協助解決問題，值得一提的是，我們在實作 MST 時需要保留下來較大的邊，等效於希望捨棄的邊總權重較小。

在最後一個作業，我選擇了兩種可能的演算法來嘗試（參考資料後附），第一種是使用 greedyFAS 演算法將頂點依照 in-degree 和 out-degree 的性質排序，

接著去掉所有的逆向邊來消去可能的迴圈。第二種則是使用 MST + DFS 的檢驗來綜合實作，其中 MST 的部分是和無向的做法相同，所以做完之後仍要處理雙向邊的部分（雙向邊亦為一迴圈），最後再將其他未使用的正邊拿進來做考慮。會多使用後者（第二種）來實作是因為發現到在 greegyFAS 中，核心的精神是盡量少移除邊，這看似很符合題目要求，不過實作後發現結果表現並沒有預期的好，那後來思考過覺得原因是有許多不影響 weakly connection 的負邊，也是存在並且可以移除來降低總權重的，因此在負邊比例高的 case，後者應會是預期可以得到較好表現的方法。

心得：

這次的作業我覺得最困難及複雜的是第三題，因為是 NP-Hard 的題目，並沒有一個一定最好的答案，因此在做之前思考了很長一段時間，也花很多時間在查找相關文獻。另一個主要會遇到的問題是出現在結果的驗證，沒有一個很有效的方式去檢查程式的正確性，例如跑出來的結果是否為 acyclic 或是 weakly connection，只能自己建構一些例子去嘗試。不過整體而言確實也是學到了之前沒有嘗試過的題目類型。

參考資料：

https://en.wikipedia.org/wiki/Feedback_arc_set

<http://www.vldb.org/pvldb/vol10/p133-simpson.pdf>