

</高二上下學期自主學習計畫

- C++自學紀錄與心得

& APCS檢定成果 />

213 18 林亭玟

</ 目錄

{01}p3-p4

主題簡介

- 動機
- 預期效益
- 進度大綱

{02}p5

成果

- APCS檢定結果證明

{03}p6-p7

心得&改善

分享自學程式的困難和心情，以及對於學習方法如何改善的反思

{04}p8-p11

計畫安排表

包括上下學期每堂自主學習課的紀錄結果

{05}p12-p14

學習資源&流程

自學用的網站、教材...

{06}p15-p36

學習內容紀錄

詳細記錄各個學習章節的練習題程式碼，並附上練習過程中的小筆記

簡介

</ 自學C++語言，利用網路上的學習教材與練習題進行，並參加APCS檢核自己的學習程度。 />

動機

未來有考慮考取資工相關科系，想預先接觸程式基礎，期望在自主學習的時間中將C++語言的程式能力培養好，同時在過程中探索自己對於程式的喜好度。

預期效益

希望能在APCS檢定取得觀念題及實作題至少3級的成績。

</ 進度大綱

高一暑假

先認識了一些C++基礎內容：基本輸入與輸出、變數、算數運算、資料型態

高二上

從暑假學過的章節銜接讀後續的:迴圈、陣列、字串、函數、遞迴等

第一次APCS

觀念題：二級
實作題：二級

第二次APCS

觀念題：三級
實作題：二級

高二下

複習一點上學期的函數與遞迴，
接續學習STL及演算法和練習
APCS觀念和實作的考古題



大學程式設計先修檢測成績證明

林亭紋
新竹市國立新竹女中
准考證號：112015213

身分證號：0200716143
檢測日期：2023年01月08日

科目	原始總分	級別
程式設計觀念題	44	第二級
程式設計實作題	140	第二級

檢測成績級別說明

程式設計觀念題 檢測人數2656人			程式設計實作題 檢測人數2622人		
級別	原始總分範圍	百分比*	級別	原始總分範圍	百分比*
五	90~100	0.0	五	350~400	0.4
四	70~89	4.6	四	250~349	3.2
三	50~69	26.4	三	150~249	52.2
二	30~49	49.8	二	50~149	26.6
一	0~29	19.1	一	0~49	37.5

* 該次檢測人數百分比（四捨五入取概數到小數第一位）

Page：1/1

申請日期：2023年06月20日



大學程式設計先修檢測成績證明

林亭紋
新竹市國立新竹女中
准考證號：112029238

身分證號：0200716143
檢測日期：2023年06月04日

科目	原始總分	級別
程式設計觀念題	64	第三級
程式設計實作題	100	第二級

檢測成績級別說明

程式設計觀念題 檢測人數3043人			程式設計實作題 檢測人數3008人		
級別	原始總分範圍	百分比*	級別	原始總分範圍	百分比*
五	90~100	2.0	五	350~400	0.3
四	70~89	32.0	四	250~349	2.5
三	50~69	35.1	三	150~249	25.9
二	30~49	22.0	二	50~149	22.5
一	0~29	8.8	一	0~49	48.8

* 該次檢測人數百分比（四捨五入取概數到小數第一位）

Page：1/1

申請日期：2023年06月20日

</ 心得

第一次嘗試自己學習程式語言並參加檢定，老實說過程真的**稱不上順利**，我認為最感到困難的點就是「**自己練習題目**」，寫zerojudge上的題目時要自己理解題目的意思和設計程式碼，常常會遇到不懂的地方，卻沒有老師能無時無刻為你解答，而當**沒有人能解說較複雜與困難的地方時真的很難有頭緒想 Outcome 程式碼怎麼寫**，只能**直接看別人程式碼的解答**，但是當有地方看不懂時又需要花大量的時間去找資料，甚至找到了還是有種**似懂非懂的困惑感**，感覺對於程式邊編寫的能力無法精熟的掌握。

雖然經過檢定後知道自己的能力尚有不足，但是在學習過程，我慢慢掌握到程式語言世界的樣貌，認識其中的語法、運作功能...，**儘管還無法靈活運用它們，但也覺得已經在自己能力範圍內把能學的都學了**，平時除了課業與學校事務要顧之外，放學也要固定練球，實在**很難再抽出時間深入鑽研程式**，所以儘管這次的計畫成果跟一開始預期的目標還差了一些，但只要想到每次在自主學習課時，程式不知道怎麼寫而感到頭痛、上網花了一大堆時間找資料的自己，就覺得**這一年來努力執行計畫的過程，已經是一個珍貴的經歷了。**

</改善？

嘗試看影片解說

覺得可以看一些網路上有的解題影片或是某些重要觀念的解釋，因為計劃期間我主要都是為了節省時間單看文字解說而已，但其實影片上能聽到的解說可能會有比較多細節，也更能加深自己的印象。

只寫題目是不夠的

比如說遇到一個不懂的練習題，而解法需要用到某個函式或函示庫，那我應該就要另外再開個頁面去針對那個函式或函式庫做些簡單的小訓練，從頭理解它的功能和用法。但是自己常常在寫題目時遇到類似情況只有上網查那個函式的文字解釋，覺得自己大概懂解法用這個函式的原因就好，卻沒有針對它進行實際的操作練習，讓我懷疑這是不是使我後續寫程式碼時難以進步、沒有頭緒的主因之一。

上學期自主計畫安排表&學習心得

週次	日期	自學內容	檢核進度	學習心得
3	111/09/15(四)	基本輸入與輸出、變數、算數運算、資料型態 +題目練習	完全達標	此部分在暑假練習過，學習過程順利。
4	111/09/22(四)	比較運算、邏輯運算、if 判斷、switch 判斷+題目練習	完全達標	此部分在暑假練習過，學習過程順利。
5	111/09/29(四)	遞增遞減運算子、迴圈 (for)、一維陣列 (array)+題目練習	完全達標	剛接觸c++迴圈與陣列，題目做得較不上手。
6	111/10/06(四)	遞增遞減運算子、迴圈 (for)、一維陣列 (array)+題目練習	完全達標	漸漸熟悉c++迴圈與陣列，但題目還是做得有點慢。
7	111/10/13(四)	迴圈 (while)、進階迴圈控制 (continue/break)+題目練習	稍有落後	因為上一部份的for迴圈+一維陣列還有點不熟，所以還沒先進這份進度。
8	111/10/20(四)	多維陣列、多重迴圈 +題目練習	稍有落後	要補學上禮拜落後的 while迴圈+進階迴圈控制，多維陣列+多重迴圈只碰了一點。
9	111/10/27(四)	多維陣列、多重迴圈 +題目練習	完全達標	接觸比較多迴圈與陣列的題目後，雖然難度越來越高，但也漸漸熟悉。
10	111/11/03(四)	函數+題目練習	完全達標	因為找到的題目不多，函數只學了比較基礎的部分。

11	111/11/10(四)	內建函數 vs. 自訂函數、全域變數 vs. 區域變數+題目練習	完全達標	因為找到的題目不多，函數只學了比較基礎的部分。
12	111/11/17(四)	C++字串(string)、常用字串函數+題目練習	稍有落後	雖然有完成進度，但沒學得很純熟。
14	111/12/01(四)	遞迴 (Recursion)+題目練習	完全達標	有完成進度大概把概念弄熟。
15	111/12/08(四)	遞迴 (Recursion)+題目練習	完全達標	有將題目完成，但題目難易度不高。
16	111/12/15(四)	C++複習+題庫練習	完全達標	複習一些之前不太熟的地方，像是迴圈與陣列不太上手的題目。
17	111/12/22(四)	C++複習+題庫練習	完全達標	嘗試做一些APCS考古題，評估自身要加強迴圈與陣列部分，並接觸了一些STL。
18	111/12/29(四)	C++複習+題庫練習	完全達標	主要練習了多維迴圈以及二維陣列的部分和做一些APCS考古題。
19	112/01/05(四)	C++複習+題庫練習	完全達標	主要練習了多維迴圈以及二維陣列的部分和做一些APCS考古題。
20	112/01/12(四)	C++複習+題庫練習	無	因期末考將至，此堂複習學校課業

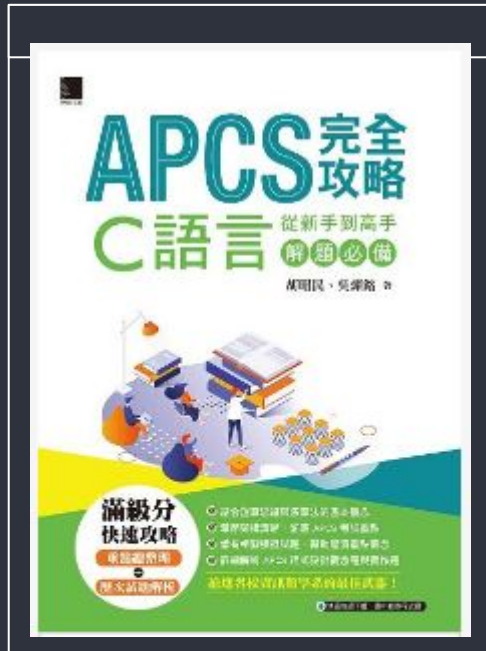
下學期自主計畫安排表&學習心得

週次	日期	自學內容	檢核進度	學習心得
1	112/02/16(四)	函數	完全達標	順利做完章節練習題。
2	112/02/23(四)	函數	完全達標	順利做完章節練習題。
3	112/03/02(四)	APCS歷屆觀念題練習	稍有落後	上網搜尋觀念題的學習資源花了大部分時間，沒練到多少題。
4	112/03/09(四)	遞迴	完全達標	為上學期學習過的內容，因為覺得需要加強，多做了相關練習題複習。
5	112/03/16(四)	遞迴	完全達標	順利完成章節練習題並提前學習STL。
6	112/03/23(四)	APCS歷屆觀念題練習	完全達標	找到105、106年APCS觀念考題內容，不會的題目找程式書相關主題加強學習。
7	112/03/30(四)	STL-vector	完全達標	完成章節練習題並提前進行下次進度queue。
8	112/04/06(四)	STL-queue	完全達標	完成章節練習題並提前用額外時間學習set、map。

9	112/04/13(四)	APCS歷屆觀念題練習	稍有落後	發現檢定需要考「演算法」內容所以學習內容改成演算法。
10	112/04/20(四)	STL-stack	無	改成學習演算法-枚舉，與做練習題，並用在家時間學習演算法-排序
11	112/04/27(四)	STL-set	完全達標	改成學習演算法-枚舉、排序的練習題。
12	112/05/04(四)	STL-map	完全達標	改成學習演算法-貪心法則，並做練習題。
13	112/05/11(四)	APCS歷屆觀念題練習	稍有落後	改成寫APCS實作考古題，挑選有演算法範圍的考古題練習演算法應用。
14	112/05/18(四)	運用C++練習APCS實作題型	完全達標	改成寫APCS實作考古題，挑選有演算法範圍的考古題練習演算法應用。
15	112/05/25(四)	運用C++練習APCS實作題型	完全達標	改成寫APCS實作考古題，挑選有演算法範圍的考古題練習演算法應用。
16	112/06/01(四)	APCS歷屆觀念題練習	完全達標	重新複習105、106年APCS觀念考題內容。
17	112/06/08(四)	運用C++練習APCS實作題型	無	APCS檢定已結束(6/4)，因此結束自主學習
18	112/06/15(四)	運用C++練習APCS實作題型	無	APCS檢定已結束(6/4)，因此結束自主學習

</ 學習資源

- C++ 基礎語法 – Yui Huang 演算法學習筆記
<https://yuihuang.com/syntax/>
- APCS(大學程式設計先修檢測)程式實作 - HackMD
<https://hackmd.io/@cubeatcysh/H1TjWAb0E?type=view#APCS%E5%A4%A7%E5%AD%B8%E7%A8%8B%E5%BC%8F%E8%A8%AD%E8%A8%88%E5%85%88%E4%BF%AE%E6%AA%A2%E6%B8%AC%E7%A8%8B%E5%BC%8F%E5%AF%A6%E4%BD%9C>
- 高中生程式解題系統
<https://zerojudge.tw/>
- 電子書: APCS完全攻略: 從新手到高手 C語言解題必備
- 其他: 網頁查詢 (Google、CSDN.....)
- 線上編輯器: Replit



</ 學習流程

1. 先看章節講義
2. 做章節範圍的練習題
->用replit篇寫程式碼及保存

Unit-5 迴圈 (while) 、進階迴圈控制 (continue/break) 【講義】

oj	題目	題解 (內含說明)
ZeroJudge	d070: 格瑞哥里的煩惱 (0 尾版)	Link
ZeroJudge	d071: 格瑞哥里的煩惱 (EOF 版)	Link
ZeroJudge	e834: P1.批發出貨(Wholesale)	Link
ZeroJudge	a148: You Cannot Pass?!	Link
ZeroJudge	d010: 盈數、虧數和完全數	Link
ZeroJudge	c067: 00591 – Box of Bricks	Link
ZeroJudge	a010: 因數分解	Link
ZeroJudge	a147: Print it all	Link
ZeroJudge	e969: 3. 大吃大喝 (Big eater)	Link
ZeroJudge	c079: 10346 – Peter's Smokes	Link

【範例】while 迴圈

Example 5-7

【Input】正整數 N

【Output】5 的倍數中，
不小於 N 的最小值


$\geq N$

0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, ...

N = 37

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int N;
6     cin >> N;
7     int ans = 0;
8     while (ans < N) {
9         ans += 5;
10    }
11    cout << ans << "\n";
12    return 0;
13 }
```

▼用replit編寫練習題的程式碼並儲存(示意圖)




林亭玟 123


@0102971


Edit Copy profile link


0 followers 0 following


Online


 **d385. 10905 - Children's Game** 1 month ago


 Public


 **f640. 函數運算式求值** 1 month ago


 Public


 **f832. 隕石 (Meteorite)** 1 month ago


 Public


 **f461. 現金兌換點卷** 1 month ago


 Public


 **j124. 3. 石窟探險** 1 month ago


 Public


 **j607. 3. 先加後乘與函數 (1)** 1 month ago


 Public


 **i401: 3. 雷射測試(待完成)** 1 month ago


 Public


 **a465. 12405 - Scarecrow(WA)** 2 months ago


 Public


 **dhafu** 2 months ago


 Public


 **能力競賽** 2 months ago


 Public


 **c292. APCS2017-0304-3數字龍捲風 (1)** 2 months ago


 Public


 **c296. APCS-2016-1029-3定時K彈** 2 months ago


 Public


 **d192. 11351 - Last Man Standing(約瑟夫環問)**


 Public


 **c036. 00573 - The Snail** 2 months ago


 Public


 **d732. 二分搜尋法 ((TLE** 2 months ago


 Public


 **f581. 3. 圓環出口((20% T_T** 2 months ago


 Public


 **c942. 露營區規劃** 2 months ago


 Public


 **f815. TOI_y21m4_a01遊戲升等** 2 months ago


 Public


 **b836. kevin戀愛攻略系列題-2 說好的霸王花呢??** 2 months ago

 Public

 **e541. 10474 - Where is the marble** 2 months ago

 Public

 **f679. 公會成員** 2 months ago

 Public

1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 1 0 1 1 0 1 1 0 1 1 1 0 1 1 1 1 0 1

遞增遞減運算子、迴圈 (for)、一維陣列 (array)

main.cpp a058: MOD3 (1)u4 0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int t;
6     cin >> t;
7
8     int x, ans[3]={};
9
10    for(int i = 0; i < t; i++){
11        cin >> x;
12        ans[x%3]++;
13    }
14    for(int i = 0; i < 3; i++){
15        cout << ans[i] << " ";
16    }
17    return 0;
18 }
```

- For迴圈
- 一維陣列

main.cpp a005: Eva 的回家作業u4 0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     int t,a,b,c,d,e;
5     cin >> t;
6     for( int i=0 ; i < t ; i++){
7         cin >> a >> b >> c >> d;
8         if(b-a==c-b){
9             e = d+(b-a);
10            cout << a << " " << b << " " << c << " " << d << " " << e << "\n";
11        }else{
12            e = d*(b/a);
13            cout << a << " " << b << " " << c << " " << d << " " << e << "\n";
14        }
15    }
16    return 0;
17 }
```

- For迴圈
- If else 條件判斷

迴圈 (while)、進階迴圈控制 (continue/break)

---c067: 00591 - Box of Bricks u5 71

main.cpp 0102971 a148: You Cannot Pass?! 標籤: 數學 u5

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     while(cin >> n){
7         int score;
8         int sum=0;
9         for(int i = 0; i < n; i++){
10             cin >> score;
11             sum = sum + score;
12         }
13         if(sum>59*n){
14             cout << "no" << "\n";
15         }else{
16             cout << "yes" << "\n";
17         }
18     }
19     return 0;
20 }
```

- For迴圈
- While迴圈
- if、else

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n, T = 1;
6
7     while(cin >> n && n){
8         int sum = 0;
9         int a[n];
10         for(int i=0; i<n; i++){
11             cin >> a[i];
12             sum += a[i];
13         }
14
15         int avg = sum / n;
16         int ans = 0;
17         for (int i=0; i<n; i++){
18             if (a[i]>avg)
19                 ans += a[i] - avg;
20         }
21         cout << "Set #" << T++ << "\n";
22         cout << "The minimum number of moves is" << ans << ".\n\n";
23     }
24     return 0;
25 }
```

每當輸入n時，
n若為0即break

多維陣列、多重迴圈

main.cpp a015: 矩陣的翻轉 u6 0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     int r, c;
5     int a[100][100];
6     while (cin >> r >> c) {
7         for (int i = 0; i < r; i++) {
8             for (int j = 0; j < c; j++) {
9                 cin >> a[i][j];
10            }
11        }
12        for (int i = 0; i < c; i++) {
13            for (int j = 0; j < r; j++) {
14                cout << a[j][i] << " ";
15            }
16            cout << "\n";
17        }
18    }
19    return 0;
20 }
```

d673: 11608 - No Problem u6 0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     int Case = 1;
7     while (cin >> n) {
8         if (n < 0) break;
9         int a[11], b[11];
10        for (int i = 0; i < 12; i++) {
11            cin >> a[i];
12        }
13        for (int i = 0; i < 12; i++) {
14            cin >> b[i];
15        }
16        cout << "Case" << Case++ << ":\n";
17        for (int i = 0; i < 12; i++) {
18            if (n > b[i]) {
19                n -= b[i];
20                cout << "No problem! :D\n";
21            }
22            else {
23                cout << "No problem. :(\n";
24            }
25            n += a[i];
26        }
27    }
28    return 0;
29 }
```

c013: 00488 - Triangle Wave u6 0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     int n;
5     while (cin >> n) {
6         int A, F;
7         for (int i = 0; i < n; i++) {
8             cin >> A >> F;
9
10            for (int i = 0; i < F; i++) {
11                for (int j = 1; j <= A; j++) {
12                    for (int k = 1; k <= j; k++) {
13                        cout << j;
14                    }
15                    cout << endl;
16                }
17                for (int j = A - 1; j > 0; j--) {
18                    for (int k = 1; k <= j; k++) {
19                        cout << j;
20                    }
21                    cout << endl;
22                }
23                cout << endl;
24            }
25        }
26    }
27    return 0;
28 }
```

```
1 #include <iostream>
2 #include <cstring>
3 using namespace std;
4
5 int main() {
6     int n;
7     while(cin >> n){
8         int pre, now;
9         int d[n]; // initial value not guaranteed to be zero
10        for(int i=1; i<n; i++){
11            d[i]=0;
```

陣列中的元素初始值不一定為0

initial value not guaranteed to be zero

```
12    }
13    cin >> pre;
14    for(int i=1; i<n; i++){
15        cin >> now;
16        d[abs(now-pre)]++;
17        pre = now;
18    }
19    bool flag = true;
20    for(int i=1; i<n; i++){
21        if(d[i] != 1){
22            flag = false;
23            break;
24        }
25    }
26    if(flag) cout << "Jolly" << "\n";
27    else cout << "Not jolly" << "\n";
28
29    return 0;
30 }
31 }
```

function 函數、內建函數 vs. 自訂函數

、全域變數 vs. 區域變數

```
a006: 一元二次方程式 u7 0102971
main.cpp x +
main.cpp > ...
1 #include <iostream>
2 #include <cmath>
3 using namespace std;
4 int main() {
5     int a, b, c;
6     cin >> a >> b >> c;
7     if(b*b-4*a*c==0){
8         cout << "Two same roots x=" << (-1*b)/2*a << "\n";
9     }
10    else if(b*b-4*a*c>0){
11        cout << "Two different roots x1=" << (-1*b+sqrt(b*b-4*a*c))/2*a
12        << " , x2=" << (-1*b-sqrt(b*b-4*a*c))/2*a << "\n";
13    }
14    else{
15        cout << "No real root\n";
16    }
17    return 0;
18 }
```

內建函數

sqrt() 是 cmath 函式庫的函數, 用於求給定數字的平方根

```
d255: 11417 - GCD u7 0102971
main.cpp x +
main.cpp > ...
1 #include <iostream>
2 using namespace std;
3
4 int GCD(int a, int b) {
5     while((a=a%b)&&(b=b%a));
6     return a+b;
7 } // 輾轉相除法
8
9 int main() {
10     int N, G;
11     while (cin >> N && N) {
12         G = 0;
13         for (int i = 1; i < N; i++) {
14             for (int j = i + 1; j <= N; j++) {
15                 G += GCD(i, j);
16             }
17         }
18         cout << G << endl;
19     }
20
21     return 0;
22 }
```

自訂函數

main.cpp × +

main.cpp > ...

```
1  #include <cmath> //把數學函數包含進去的一個頭文件命令
2  #include <iostream>
3  using namespace std;
4
5  int main() {
6      int t;
7      double n, m;
8      cin >> t;
9      while (t--) {
10         cin >> n >> m;
11         cout << (int)ceil((n - 2) / 3) * (int)ceil((m - 2) / 3) << endl;
12     }
13     // while(N--) 的意思是執行while循環,每執行一次{} N減1 ,直到N=0時跳出循環
14     //邊緣的格子不需要監控,所以長邊短邊都是先減去 2,再除以 3 後無條件進位
15     // ceil( )無條件向上取整,但其回傳值資料型態為 double
16     return 0;
17 }
```

C++字串(string)、常用字串函數

`</ getline() />`

getline()的用法:
接收一個字串, 可以接收
空格並輸出

再一次呼叫getline()把
緩衝區清空, 因為換行
符(\n)是此函數的有效輸
入

1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1

main.cpp

main.cpp > ...

```
1 #include <cmath>
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     int t;
7     string s;
8     cin >> t;
9     getline(cin, s); //把緩衝區清空
10    while (t--) {
11        getline(cin, s);
12        int n = sqrt(s.size());
13        if (n * n != s.size()) {
14            cout << "INVALID\n";
15            continue; //執行continue時C++會馬上跳回迴圈的開頭然後再繼續執行
16        }
17
18        for (int col = 0; col < n; col++) {
19            for (int row = 0; row < n; row++) {
20                cout << s[row * n + col];
21            }
22        }
23        cout << endl;
24    }
25    return 0;
26 }
```


stringstream是一個C++底下的類別，專門拿來讀取字串並且處理，很多時候拿來做字串的切割，或者是int跟string類別之間的轉換

main.cpp

main.cpp > f main

```
1  #include <iostream>
2  #include <sstream>
3  using namespace std;
4  int main() {
5      int t;
6      string s;
7      char head,tail;
8      cin>>t;
9      getline(cin,s);
10 while(t--){
11     bool loop = true;
12     int len = 0; //軌道個數
13
14     getline(cin,s);
15
16     if (s[0] == s[s.size()-1]){
17         loop = false;
18     }
19
20     else{
21         stringstream ss(s);
22         ss >> s;
```

```
23         len = 1;
24         tail = s[s.size()-1];
25         while(ss>>s){
26             len++;
27             head = s[0];
28             if(head == tail){
29                 loop = false;
30                 break;
31             }
32             tail = s[s.size()-1];
33         }
34     }
35     if (len>1&&loop) cout<<"LOOP\n";
36     else cout<<"NO LOOP\n";
37 }
38 return 0;
39 }
```

1 0 1 1 0 1 1 0 1 1 0 1

1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 1

遞迴 (Recursion)

c813: 11332 - Summing Digits u10
vera20069544

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int f(int x){
5     int y=0;
6     while(x>0){
7         y += x%10;
8         x = x/10;
9     }
10    if(y<10) return y;
11    else return f(y);
12 }
13
14
15 int main() {
16     int n;
17     while (cin >> n && n) {
18         cout << f(n) << endl;
19     }
20     return 0;
21 }
```

$f(n)$ 為 n 的每一個
十進位數字的總和

1 0 1

1 1

c002: 10696 - f91 u10
vera20069544

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int f91(int N){
5     if(N<=100){
6         return f91(f91(N+11));
7     }else{
8         return N-10;
9     }
10 }
11
12 int main() {
13     int N,y;
14     while(cin >> N && N){
15         y = f91(N);
16         cout << "f91(" << N << ") = " << y << endl;
17     }
18     return 0;
19 }
```

如果 $N \leq 100$, 那麼 $f91(N)$
= $f91(f91(N+11))$
.
如果 $N \geq 101$, 那麼 $f91(N)$
= $N-10$

1

STL 標準模板庫

map: 存放 key-value pairs 的映射資料結構, 會按 key 由小到大排序

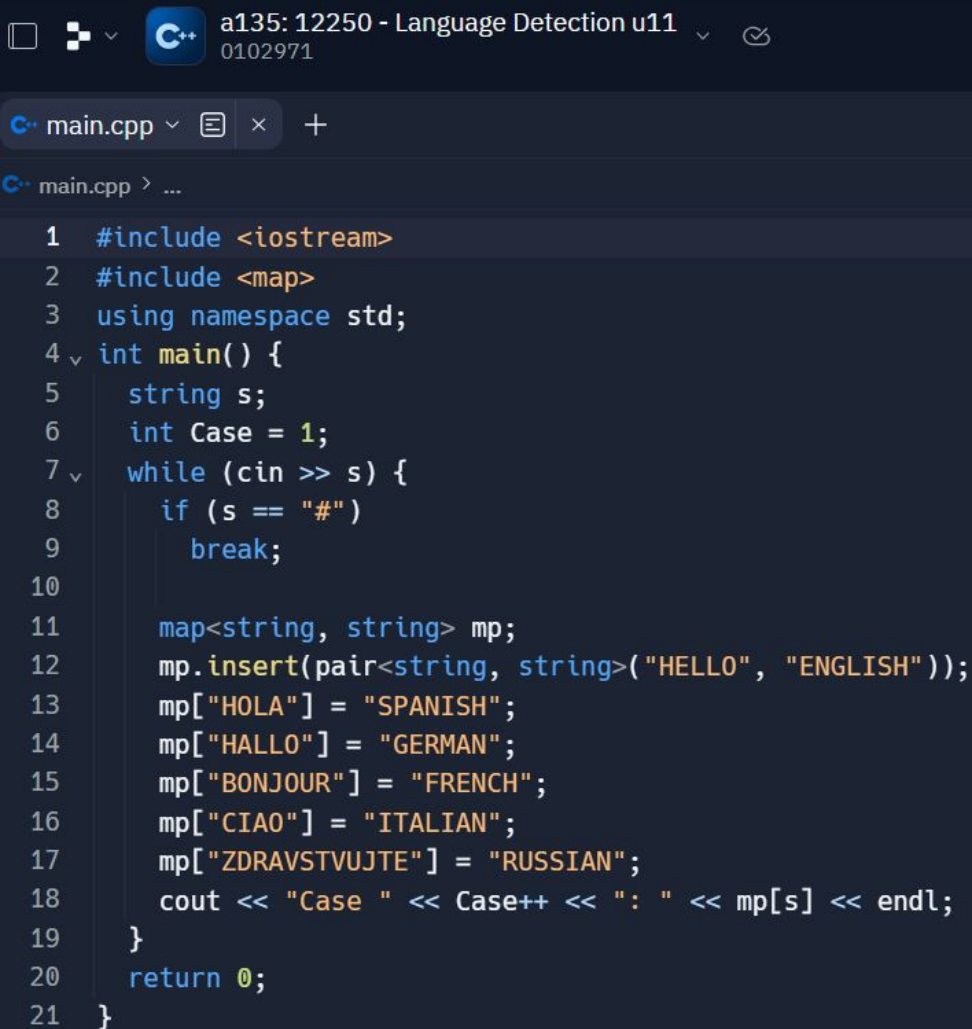
- 加入新的key-value pair

mp[key] = value

- 利用 key 取值:

value = mp[key];

1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1



The screenshot shows a C++ IDE with a file named 'main.cpp'. The code defines a map 'mp' that maps words to their translations. It then reads a word from the user and prints its translation from the map.

```
1 #include <iostream>
2 #include <map>
3 using namespace std;
4 int main() {
5     string s;
6     int Case = 1;
7     while (cin >> s) {
8         if (s == "#")
9             break;
10
11         map<string, string> mp;
12         mp.insert(pair<string, string>("HELLO", "ENGLISH"));
13         mp["HOLA"] = "SPANISH";
14         mp["HALLO"] = "GERMAN";
15         mp["BONJOUR"] = "FRENCH";
16         mp["CIAO"] = "ITALIAN";
17         mp["ZDRAVSTVUJTE"] = "RUSSIAN";
18         cout << "Case " << Case++ << ": " << mp[s] << endl;
19     }
20     return 0;
21 }
```


main.cpp

main.cpp > f main

```
1 #include <iostream>
2 #include <set>
3 using namespace std;
4
5 int main() {
6     set<int> st;
7     int t;
8     int Case = 1;
9     while (cin >> t && t) {
10         int v[t];
11         for (int i = 0; i < t; i++) {
12             cin >> v[i];
13         }
14         int a;
15         bool an = true;
16         for (int i = 0; i < t; i++) {
17             for (int j = i; j < t; j++) {
18                 a = v[i] + v[j];
19                 if (st.count(a)) {
20                     an = false;
21                     break;
22                 } else {
```

Set: 集合, 去除重複的元素, 資料由小到大排序

- 把元素 x 加進 set:

`st.insert(x)`

- 檢查元素 x 是否存在 set 中:

`st.count(x);`

```
22 } else {
23     st.insert(a);
24 }
25 }
26 if (!an) {
27     break;
28 }
29 }
30
31 if (an) {
32     cout << "Case #" << Case++ << ": It is a B2-Sequence." << endl;
33 } else {
34     cout << "Case #" << Case++ << ": It is not a B2-Sequence." << endl;
35 }
36 cout << endl;
37 }
38 return 0;
39 }
```

queue: 佇列, 資料有「先進先出」(first in first out, FIFO) 的特性

- 把元素 x 加進 queue:

q.push(x);

- 取值:

x = q.front();

- 移除已經讀取的值:

q.pop()

1 0 1 1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1

```
e155: 10935 - Throwing cards away I u11  
vera20069544  
  
main.cpp  
main.cpp > ...  
1 #include <iostream>  
2 #include <queue>  
3 using namespace std;  
4  
5 int main() {  
6     int t;  
7     while (cin >> t && t) {  
8         queue<int> q;  
9         for (int i = 1; i <= t; i++) {  
10             q.push(i);  
11         }  
12         cout << "Discarded cards: ";  
13         while (q.size() != 1) {  
14             cout << q.front() ;  
15             if (q.size() != 2)  
16                 cout << ", ";  
17             q.pop();  
18             q.push(q.front());  
19             q.pop();  
20         }  
21         cout << endl << "Remaining card: " << q.front() << endl;  
22     }  
23     return 0;  
24 }
```

演算法-枚舉

b230. TOI2009 第二題：方便數
0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     const int maxn = 1850;
6     int n, a[maxn], tmp, i = 1, j = 2, k = 3, cnt = 0;
7     while (cin >> n) {
8         while (1) {
9             while (1) {
10                 while (1) {
11                     tmp = i * j + j * k + k * i;
12                     if (tmp >= maxn)
13                         break;
14                     a[tmp] = 1;
15                     k++;
16                 }
17                 j++;
18                 k = j + 1;
19                 tmp = i * j + j * k + k * i;
20                 if (tmp > maxn)
21                     break;
22             }
23             i++;
24             j = k + 1;
25             k = j + 1;
26             tmp = i * j + j * k + k * i;
27             if (tmp > maxn)
28                 break;
29         }
30         for (int i = 1; i < maxn; i++) {
31             if (a[i] == 0) {
32                 cnt++;
33                 if (cnt == n) {
34                     cout << i << endl;
35                     break;
36                 }
37             }
38         }
39         cout << endl;
40     }
41     return 0;
42 }
```

```
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
```

演算法-排序

1 0 1 1 0 1 1 0 1

d150. 11369 - Shopaholic 排序
0102971

```
main.cpp × +  
main.cpp  
1 #include <iostream>  
2 #include <algorithm>  
3 #include <vector>  
4 using namespace std;  
5  
6 int main() {  
7     int t,n;  
8     while(cin>>t>>n){  
9         int p[n];  
10        for(int i=0;i<n;i++){  
11            cin>>p[i];  
12        }  
13  
14        sort(p,p+n);  
15        reverse(p,p+n);  
16  
17        int sum=0;  
18        for(int i=2;i<n;i+=3){  
19            sum+=p[i];  
20        }  
21  
22        cout<<sum<<endl;  
23    }  
24    return 0;  
25 }
```

a737. 10041 - Vito's family 排序
0102971

```
main.cpp × +  
main.cpp > ...  
1 #include <iostream>  
2 #include <algorithm>  
3 using namespace std;  
4  
5 int main() {  
6     int r;  
7     cin>>r;  
8     while(r--){  
9         int c,sum=0;  
10        cin>>c;  
11        int s[c];  
12        for(int i=0;i<c;i++){  
13            cin>>s[i];  
14        }  
15        sort(s,s+c);  
16        for(int i=0;i<c;i++){  
17            sum+=abs(s[i]-s[c/2]);  
18        }  
19        cout<<sum<<endl;  
20    }  
21    return 0;  
22 }
```

main.cpp

main.cpp > f main

```

1  #include <algorithm>
2  #include <cstring>
3  #include <iostream>
4  using namespace std;
5
6  int main() {
7      ios_base::sync_with_stdio(0);
8      cin.tie(0);
9      string s1, s2;
10     int t, x;
11     while (cin >> s1 >> s2) {
12         cin >> t;
13         int a[26];
14

```

```

14
15     memset(a, -1, sizeof(a));
16     // memset(结构体/数组名, "用于替换的字符", 前n个字符) => 将a全设为-1;
17
18     for (int i = 0; i < s1.size(); i++) {
19         a[s1[i] - 'A'] = i;
20     }
21
22     int len = s2.size();
23     pair<int, char> ans[len];
24     for (int i = 0; i < len; i++) {
25         ans[i] = {a[s2[i] - 'A'], s2[i]};
26     }
27     sort(ans, ans + len);
28
29     for (int i = 0; i < t; i++) {
30         cin >> x;
31         cout << ans[x - 1].second << endl;
32     }
33 }
34 return 0;
35 }

```

可以增加cin, cout
效率, 節省時間

演算法-模擬

d192. 11351 - Last Man Standing(約瑟夫環問題 Josephus)
0102971

main.cpp

main.cpp > f main

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     cin.sync_with_stdio(false);
6     cin.tie(nullptr);
7     int t, Case=1;
8     cin >> t;
9     while (t--) {
10         int n, k, s = 0;
11         cin >> n >> k;
12         for (int i = 2; i <= n; i++) {
13             s = (s + k) % i; // s(i=1)==0
14         }
15         cout << "Case " << Case++ << ": " << s+1 << "\n";
16     }
17     return 0;
18 }
```

c036. 00573 - The Snail 模擬
0102971

main.cpp

main.cpp > f main

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     double h,u,d,f,reduce;
6     while(cin>>h>>u>>d>>f){
7         if(h==0) break;
8         double sum=0;
9         reduce=u*0.01*f;
10        int day=0;
11        while(1){
12            day++;
13            sum=sum+u;
14            if(sum>h) break;
15            sum=sum-d;
16            if(sum<0) break;
17            if(u>0)u=u-reduce;
18            else u=0;
19        }
20        if(sum>=h) cout<<"success on day "<<day<<"\n";
21        else cout<<"failure on day "<<day<<"\n";
22    }
23    return 0;
24 }
```


演算法-貪心法則

e538. 11389 - The Bus Driver Problem 貪心
0102971

main.cpp

main.cpp > ...

```
1  #include <iostream>
2  #include <algorithm>
3  using namespace std;
4
5  int cmp(int x,int y){
6      return x>y;
7  }
8
9  int main() {
10     int n,d,r;
11     while(cin>>n>>d>>r){
12         if(n==0) return 0;
13         int mor[n],nig[n];
14         for(int i=0;i<n;i++){
15             cin>>mor[i];
16         }
```

```
17     for(int i=0;i<n;i++){
18         cin>>nig[i];
19     }
20     sort(mor,mor+n);
21     sort(nig,nig+n,cmp);
22
23     int fare =0;
24     for(int i=0;i<n;i++){
25         if((mor[i]+nig[i]-d)>0){
26             fare+=(mor[i]+nig[i]-d)*r;
27         }
28     }
29     cout<<fare<<endl;
30 }
31 return 0;
32 }
```

1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1

APCS歷屆實作題

i400: 2. 字串解碼 APCS2
0102971

main.cpp

main.cpp > ...

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     int m, n;
7     cin >> m >> n;
8     string e[m];
9     for(int i=0; i<m; i++){
10         cin >> e[i];
11     }
12     string S="", T="";
13     cin >> T;
14     for(int i=m-1; i>=0; i--){
15         int count_1=0;
16         S = "";
17         for(int j= n-1; j>=0; j--){
18             if(e[i][j]=='1'){
19                 S = S + T[j]; // "1": S最後一個 <-> T最後一個
```

```
20         count_1 += 1;
21     }else{
22         S = T[j] + S; // "0": S第一個 <-> T最後一個
23     }
24 }
25 if(count_1%2==1){
26     if(n%2==1){
27         S = S.substr(n/2+1,n/2) + S.substr(n/2,1) + S.substr(0,n/2);
28     }else{
29         S = S.substr(n/2,n/2) + S.substr(0,n/2);
30     }
31 }
32 T = S;
33 }
34 cout << S << endl;
35 return 0;
36 }
```

1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 1 0 1 1 1 1 1 0 1

APCS歷屆實作題

```
f640. 函數運算式求值 APCS 0102971
main.cpp
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main() {
5     string s;
6     stack<string> stk1;
7     stack<int> stk2;
8     while (cin >> s) {
9         stk1.push(s);
10    }
11    while (!stk1.empty()) {
12        s = stk1.top();
13        stk1.pop();
14        if (s == "f") {
15            int x = stk2.top();
16            stk2.pop();
17            stk2.push(2 * x - 3);
18        } else if (s == "g") {
19            int x = stk2.top();
20            stk2.pop();
21            int y = stk2.top();
22            stk2.pop();
23            stk2.push(2 * x + y - 7);
24        } else if (s == "h") {
25            int x = stk2.top();
26            stk2.pop();
27            int y = stk2.top();
28            stk2.pop();
29            int z = stk2.top();
30            stk2.pop();
31            stk2.push(3 * x - 2 * y + z);
32        } else {
33            stk2.push(stoi(s));
34        }
35    }
36    cout << stk2.top() << endl;
37    return 0;
38 }
```

```
j124. 3. 石窟探險 APCS 0102971
main.cpp
6 int dft(){
7     int n;
8     cin>>n;
9     if (n==0) return 0;
10    else if(n%2==0){
11        int l=dft(),r=dft();
12        if(l) ans+=abs(n-l);
13        if(r) ans+=abs(n-r);
14    }
15    else if(n%2){
16        int l=dft(),m=dft(),r=dft();
17        if(l) ans+=abs(n-l);
18        if(m) ans+=abs(n-m);
19        if(r) ans+=abs(n-r);
20    }
21    return n;
22 }
23
24 int main() {
25     dft();
26     cout<<ans<<endl;
27     return 0;
28 }
```

1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1

APCS歷屆實作題

j607. 3. 先加後乘與函數 (1) APCS
0102971

main.cpp × +

main.cpp > f main

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 long long addmul(string s) {
5     stringstream ss;
6     ss.str(s);
7
8     long long n;
9     char c;
10    stack<long long> st;
11    ss >> n;
12    st.push(n);
13    while (ss >> c >> n) {
14        if (c == '+') {
15            st.top() += n;
16        } else if (c == '*') {
17            st.push(n);
18        }
19    }
20
21    while (st.size() > 1) {
22        n = st.top();
23        st.pop();
24        st.top() *= n;
25    }
26 }
```

```
27     return st.top();
28 }
29
30 long long f(string s) {
31     stringstream ss(s);
32     vector<long long> v;
33
34     while (getline(ss, s, ',')) { //切割字串
35         long long num = addmul(s);
36         v.push_back(num);
37     }
38     long long mx = v[0];
39     long long mn = v[0];
40     for (int i = 0; i < v.size(); i++) {
41         mx = max(mx, v[i]);
42         mn = min(mn, v[i]);
43     }
44     return mx - mn;
45 }
46
47
48 int main() {
49     string s;
50     cin >> s;
51
52     stack<int> st;
53     for (int i = 0; i < s.size(); i++) {
```

```
54     if (s[i] == 'f') {
55         st.push(i);
56     } else if (s[i] == ')') {
57         int left = st.top(), right = i;
58         st.pop();
59         string subs = s.substr(left + 2, right - left + 1 - 3);
60         // .substr(起始值, 長度) => 取字串函數
61         long long num = f(subs);
62         s.replace(left, right - left + 1, to_string(num));
63         i = left;
64     }
65 }
66
67 cout << addmul(s);
68 return 0;
69 }
```

1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 1

APCS歷屆實作題

□ ❏ ❏ b966. 第 3 題 線段覆蓋長度 0102971

main.cpp × +

main.cpp > f main

```
1 #include <algorithm>
2 #include <iostream>
3 #include <vector>
4 using namespace std;
5
6 int main() {
7     int t, L, R;
8     vector<pair<int, int>> v;
9     while (cin >> t) {
10         for (int i = 0; i < t; i++) {
11             cin >> L >> R;
12             //把線段的開始端點和結束端點視為獨立的event
13             //在線段的開始端點L處，event +1
14             v.push_back({L, 1});
15             //在線段的結束端點R處，event -1
16             v.push_back({R, -1});
17         }
18         sort(v.begin(), v.end());
19         // now指向目前檢視的event
```

```
20         // event為該座標點i，對應一個左閉右開區間[i, i+1)，共被幾段線段覆蓋
21         int leng = 0, now = 0, ans = 0;
22         for (int i = 0; i < 1e7; i++) {
23             while (now < v.size() && v[now].first == i) {
24                 leng += v[now].second;
25                 now++;
26             }
27             if (leng > 0) {
28                 ans++;
29             }
30         }
31         cout << ans << endl;
32     }
33     return 0;
34 }
```

1 1 0 1 1 0 1 1 0 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 1

APCS歷屆實作題

g597. 3. 生產線
vera20069544

main.cpp

main.cpp

```
1  #include <bits/stdc++.h>
2  using namespace std;
3
4  long long n,m,l,r,t,hour[200005],mcnum[200005],ans=0;
5
6  bool cmp(int x,int y){
7      return x>y;
8  }
9
10 int main() {
11     cin.sync_with_stdio(false);
12     cin.tie(0);
13
14     cin>>n>>m;
15     while(m--){
16         cin>>l>>r>>t;
17         l--;
18         r--;
19         mcnum[l]+=t;
20         mcnum[r+1]-=t;
21     }
22     for(long long i=1;i<n;i++){
```

```
20         mcnum[r+1]-=t;
21     }
22     for(long long i=1;i<n;i++){
23         mcnum[i]+=mcnum[i-1];
24     }
25
26     for(long long i=0;i<n;i++){
27         cin>>hour[i];
28     }
29     sort(mcnum,mcnum+n,cmp);
30     sort(hour,hour+n);
31     for(long long i=0;i<n;i++){
32         ans+=mcnum[i]*hour[i];
33     }
34     cout<<ans<<endl;
35     return 0;
36 }
```

1 1 0 1 1 0 1 1 1 0 1 1 0 1 1 1 1 1 0 1



感謝觀看!!

END