My Notes
Important Concepts worth keeping

Today: / /

單元一 遞迴

1. 反向印出字串

```
void writeBackward(string s, int size){ 法1
    if(size>0){
        cout<<s.substr(size-1,1);
                    開始位置 長度
        writeBackward(s,size-1);
    }
}
```

→較快產生第一个輸出

```
                                    位入值=0
void writeBackward(string s,int size,int start){ 法2
    if(start<=size-1){
        writeBackward(s,size,start+1);
        cout<<s.substr(start,1);
    }
}
```

→較慢產生第一个輸出

Punctuality: Showing esteem for others by doing the right things at the right time.
- (IBLP)

My Questions
Problems & Difficulties needing exploration

2. 連續整數加總(結束=遞迴)

```
int sum(int a, int b){ 法1
    if(a>b){
        return sum(a-1,b)+a;
    }
    else{          //a==b
        return a;
    }
}
```

→由小加到大

```
int sum(int a, int b){ 法2
    if(b<a){
        return sum(a,b+1)+b;
    }
    else{          //b==a
        return b;
    }
}
```

→由大加到小

My Opinions
Thoughts, inspirations, and suggestions

守時：在對的時間，做對的事，來表明對別人的尊重。
- 《培基》

My Notes
Important Concepts worth keeping

Today: / /

5 找最大值(未排序)

```
if (anArray has only one item)
    maxArray(anArray) is the item in anArray
else if (anArray has more than one item)
    maxArray(anArray) is the maximum of
    maxArray(left half of anArray) and
    maxArray(right half of anArray)
```

Faith, Hope and Love !
Faith lays the foundation,
Hope brings sunshine,
Love changes everything.
-Sundye

My Questions
Problems & Difficulties needing exploration

6 找第k小的數值(未排序)

用一個 first, last, pivotIndex 與 pivot item 在所有的陣列中排行第x

小於的放左邊，大於等於的放右邊

kSmall(k, anArray, first, last) → 在這區的陣列中排行第k

→ 這區陣列的index範圍

```
if (k < pivotIndex - first + 1)
    return kSmall(k, anArray, first, pivotIndex-1)
else if (k == pivotIndex - first + 1)
    return p
else          // k > pivotIndex - first + 1
    return kSmall(k-(pivotIndex - first + 1), anArray,
                  pivotIndex+1, last)
```

My Opinions
Thoughts, inspirations, and suggestions

陽光達人信望愛，
信實可靠打造基礎，
樂觀希望散發正能量，
真心關愛可以改變一切。
- 三采

密碼 cipher

My Notes
Important Concepts worth keeping

Today : / /

7. 反轉陣列

```
ReverseArray(anArray, low, high)
    if (low < high)
        Swap anArray[low] and anArray[high]
        ReverseArray(anArray, low+1, high-1)
```

ONCE, all the villagers decided to pray for rain. On the day of prayer all the people gathered, but only one boy came with an umbrella.
That's FAITH!

-Anonymous

My Questions
Problems & Difficulties needing exploration

8 河內塔

```
solveTowers(count, source, destination, spare)
    if (count == 1)
        Move a disk directly from source to destination
    else
        solveTowers(count-1, source, spare, destination)
        solveTowers(1, source, destination, spare)
        solveTowers(count-1, spare, destination, source)
```

My Opinions
Thoughts, inspirations, and suggestions

* moves(n) = 2moves(n)+1 = $2^n - 1$

Base case: moves(1) = 1

村人聚集祈雨，獨有一個孩子帶著雨傘前來。
這就是信心！
-佚名

My Notes
Important Concepts worth keeping

Today / /

```
畫刻度尺(二元遞迴)
void drawRuler(int nInches, int majorLength){
    drawOneTick(majorLength, 0);
    for(int i=1; i<=nInches; i++){
        drawTicks(majorLength-1);
        drawOneTick(majorLength, i);
    }
}
void drawTicks(int tickLength){
    if(tickLength>0){
        drawTicks(tickLength-1);
        drawOneTick(tickLength, -1);
        drawTicks(tickLength-1);
    }
}
```

EVERY night we go to bed, without any assurance of being alive the next morning, but still we set the alarms.
That's HOPE!

-Anonymous

My Questions
Problems & Difficulties needing exploration

```
void drawOneTick(int tickLength, int tickLabel){
    for(int i=0; i<tickLength; i++)
        cout<<"-";
    if(tickLabel>=0)
        cout<<" "<<tickLabel;
    cout<<endl;
}
```

My Opinions
Thoughts, inspirations, and suggestions

即便不知道明天如何，我們仍然照常設定鬧鐘。
這就是**希望**！

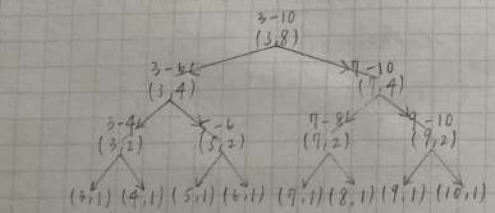
-佚名

My Notes
Important Concepts worth keeping

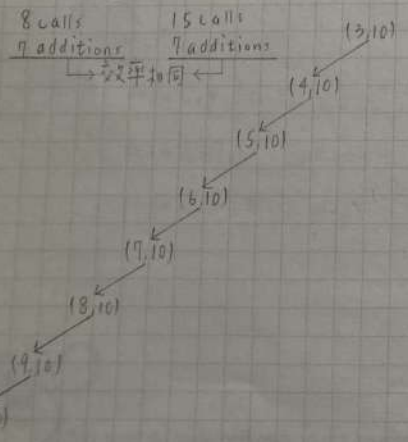
Today: / /

10 連續整數加總(二元遞迴)

```
int sum(int a, int n){          //n=b-a+1
    if(n==1)    return a;
    return sum(a,n/2)+sum(a+n/2, n-n/2);
```



★線性遞迴 vs 二元遞迴



WE see the world suffering, but still we get married.
That's LOVE! -Anonymous

My Questions
Problems & Difficulties needing exploration

11. 兔子數量(費氏數列)

(1) 兔子不會死

(2) 兔子從第3個月開始每個月都會生1對小兔

```
int rabbit(int n){
    if(n<=2)   return 1;
    else   return rabbit(n-1)+rabbit(n-2);
}

int call(int n){
    if(n<=2)   return 1;
    else   return call(n-1)+call(n-2)+1;
}
```

My Opinions
Thoughts, inspirations, and suggestions

線性遞迴

```
linearFibonacci(k){
    if(k==1)   return(k,0)
                      F1 F0
    else{
        (i,j)=linearFibonacci(k-1);
        return(i+j, i);
          Fk=Fk-1+Fk-2; Fk-1
    }
}
```

即便世態艱難，我們仍然選擇結伴同行，
這就是愛！
-佚名

My Notes
Important Concepts worth keeping

Today: , ,

3 最大公因數

法1: gcd(x, y) = x if y = 0
gcd(x, y mod x) if y > x
gcd(y, x mod y) otherwise

法2: gcd(x, y) = y if x mod y = 0 ⇒少呼叫一次
gcd(x, y) = (y, x mod y) otherwise ⇒如果 x<y，執行時會使兩者交換，多呼叫一次

總結: 如果 x ≥ y，法2較有效率
如果 x < y，兩者效率相同

We are convinced that love is the principal guiding force in education. We, teachers and students alike, pursue mutual growth through instruction by both words and deeds, in a spirit of love and respect for one another.

-《CYCU Education Philosophy》

My Questions
Problems & Difficulties needing exploration

4 二元搜尋(已排序)

```
int binarySearch( int anArray[], int first, int last, int value){
    int index;
    if (first > last){
        index = -1;   //值不存在
    }
    else{
        int mid = (first + last)/2
        if (value == anArray[mid]){
            index = mid;
        }
        else if (value < anArray[mid]){
            index = binarySearch(a, ", mid-1, "){ 左邊
        }
        else{    // value > anArray[mid]
            index = binarySearch(a, mid+1, ", "){ 右邊
        }
    return index;
}
```

My Opinions
Thoughts, inspirations, and suggestions

我們確信『愛』是教育的主導力量，願以身教言教的方式，互愛互敬的態度，師生共同追求成長。

-《中原大學教育理念》

密碼 cipher

My Notes
Important Concepts worth keeping

Today: / /

14. returnValue的平方最接近但低於n

```
int getValue(int a, int b, int n){
    int c=(a+b)/2;
    if(c*c<=n)&&(n<(c+1)*(c+1))
        returnValue=c;
    else if(c*c>n)
        returnValue=getValue(a,c-1,n);
    else                    //(c+1)*(c+1)<=n
        returnValue=getValue(c+1,b,n);
    return returnValue;
}
```

Only faith can let the thoughts spark, only hope can light up the future.
-Victor Marie Hugo

My Questions
Problems & Difficulties needing exploration

15. n取k的組合數

$C(n,k)=C(n-1,k-1)+C(n-1,k)$
（必選某物　必不選某物）

Base case: $C(n,n)=1$

$C(n,0)=1$

* The number of recursive calls to base cases is $C(n,k)$

non-base cases $C(n,k)-1$

My Opinions
Thoughts, inspirations, and suggestions

只有信心才讓思想發出火花，只有希望才讓未來發出光芒。
-雨果

My Notes
Important Concepts worth keeping

Today: / /

單元二 抽象化

1.類別

(1) Each class definition is placed in a header file.

eg

```
/** @file Sphere.h */
class Sphere
{
public:
  Sphere();                    // Default constructor
  Sphere(double newRadius);    // Constructor
  methods(return type+ method name+ const)
                                  ↑ can't change data member
private:
  double theRadius;  // data member
};
```

Suffering produces character, and character produces hope, and hope does not disappoint us.

-Jeremy Lin, NBA Player

My Questions
Problems & Difficulties needing exploration

(2) The implementation of a class' methods are placed in an implementation file

eg

```
/** @file Sphere.cpp */
#include "Sphere.h"
Sphere::Sphere(){
  theRadius=1.0
}
Sphere::Sphere(double initialRadius){
  theRadius=initialRadius;
}
```

My Opinions
Thoughts, inspirations, and suggestions

苦難造就品格，品格創造希望，而希望不會讓我們失望。

-NBA球員林書豪

My Notes
Important Concepts worth keeping

Today: / /

2. 繼承: A derived class or subclass inherits any of the publicly defined methods or data members of a base case or superclass

eg.

```
#include "Sphere.h"
enum Color{ }
class ColoredSphere : public Sphere
{
public:
  methods
private:
  Color c;
}
```

(子類別 / 父類別)

* ColoredSphere無法直接使用Sphere的 theRadius

solution: 將 private 改成 protected

* public: any class instances

protected: subclass instances

private: only class instances

If you shed tears when you miss the sun, you also miss the stars.
–Rabindranath Tagore

My Questions
Problems & Difficulties needing exploration

3. Overload 多載: 在一個類別中定義多個名稱相同但參數不同的方法

```
class Chef{
  void cook(){ //煮一份牛排
    牛排 ⇒ cook('A');
  }
  void cook(char meal){ //煮一份牛排或豬排
    if(meal=='A') 牛排
    if(meal=='B') 豬排
                  ⇒ cook(meal,1);
  }
  void cook(char meal, int quantity){ //煮n份牛排或豬排
    if(quantity==0) return;
```

My Opinions
Thoughts, inspirations, and suggestions

```
    else{
      if(meal=='A') 牛排
      if(meal=='B') 豬排
    }
    cook(meal, quantity-1);
  }
}
```

密碼 cipher

當你為了錯過太陽而哭泣，那你也將錯過群星。
–泰戈爾

My Notes
Important Concepts worth keeping

Today: / /

4. Override覆寫：子類別可以覆寫父類別的方法內容，使該方法行為不同於父類別的行為

```
class Mouse{
  void call(){
    zizichacha
  }
}
class Pikachu:Mouse{
  void pikachucall(){ => call()
    pikapika
  }
}
class Hamutaro:Mouse{
  void hamutarocall(){ => call()
    hoke
  }
}
```

Patience is the road to understanding, which is the key to a happy heart.
– 《Friends》

My Questions
Problems & Difficulties needing exploration

```
void testMouse(Mouse mouse){
  if(mouse->getType==Pikachu) mouse.pikachucall();
  else if(mouse->getType==Hamutaro) mouse.hamutarocall(); => mouse.call();
  else mouse.call();
}
```

My Opinions
Thoughts, inspirations, and suggestions

耐心是通往理解必經的路，也是擁有快樂的關鍵。
– 《六人行》

My Notes
Important Concepts worth keeping

Today: / /

5 命名空間

Creating a namespace:

```
namespace small{
  int count=0
  void abc();
}
```

Using a namespace.

```
using namespace small{
  count+=1;
  abc();
}
```

* std為標準函式庫，using namespace std 能使std內定義的識別符號都有效，好像它們被宣告為全域變數一樣，不用在前面加std::

Every cloud has a silver lining.
-Gilbert

My Questions
Problems & Difficulties needing exploration

6 例外狀況

try 設定保護範圍

catch 補捉例外狀況

eg input<0就印出錯誤訊息

```
int i;
cin>>i;
try{
  if(i<0) throw "something wrong";
}
catch(const char* message){
  cout<<message<<endl;
}
```

(break越到catch)

My Opinions
Thoughts, inspirations, and suggestions

每朵烏雲背後都有陽光。
-西伯特

My Notes
Important Concepts worth keeping

Today: / /

單元二 鏈結串列

1. Pointer

(1) Declare of an integer pointer variable p: int* p;

(2) The address-of operator &: p=&x;

(3) The new operator: p=new int;

(4) delete p;

p=Null // safeguard

※如順序不可對調否則會造成 memory leak

You must not let anyone define your limits because of where you come from. Your only limit is your soul.
- 《Ratatouille》

My Questions
Problems & Difficulties needing exploration

2. Comparing array based and pointer based implementation

	Array	Pointer
storage requirement if no waste	✓	
retrieval	✓ Constant	Depends on i
insertion and deletion	Shifting	✓ Traversal

My Opinions
Thoughts, inspirations, and suggestions

別讓任何人用出身定義你，你唯一的限制只有你的靈魂。
- 《料理鼠王》

My Notes
Important Concepts worth keeping

Today: / /

12 指數函數

```
double power(double x, int n){          //法1
    if(n==1)    return x;
    else    return x*power(x,n-1);
}
double power(double x, int n){
    if(n==0)    return 1;
    else{
        double halfpower = power(x, n/2)
        if(n%2==0)    return halfpower*halfpower        //n is even
        else    return x*halfpower*halfpower            //n is odd
    }
}
```

乘法次數：　q^{32}　q^{19}　呼叫次數　q^{32}　q^{19}

法1　32　19　　法1　32　19

法2　9　8　　法2　9　6

(1) (1) (1) (1) (1) (2) (0)　(2) (2) (1) (1) (2) (0)

32→16→8→4→2→1→0 、19→9→4→2→1→0

To accomplish great things, we must not only act, but also dream, not only plan, but also believe.

-Anatole France

My Questions
Problems & Difficulties needing exploration

13 遊行隊伍排列數

(1) 遊行由樂隊和花車組成。
P　B　F

(2) 樂隊不能緊跟著另一個樂隊
花車　樂隊
P(n) = F(n) + B(n)
前一個　前二個

F(n) = P(n-1)

B(n) = F(n-1) = P(n-2)

⇒ P(n) = P(n-1) + P(n-2)　for n > 2

Base case: P(1) = 2 (F, B)

P(2) = 3 (FF, BF, FB)

My Opinions
Thoughts, inspirations, and suggestions

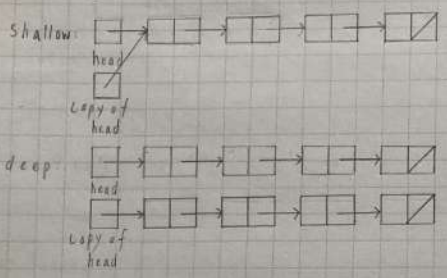
為成就大事，我們需要的不只是行動，還要夢想，不只是計畫，還要信心。

-阿納托爾·法郎士

My Notes
Important Concepts worth keeping

Today: / /

3 Comparing shallow copy and deep copy of linked lists

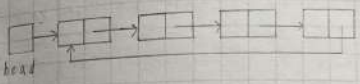


We all deserve a standing ovation at least once in our lives.
- 《Wonder》

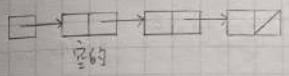
My Questions
Problems & Difficulties needing exploration

4 Variations of linked lists

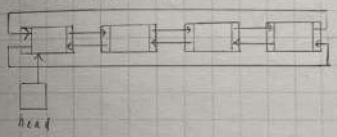
(1) Circular linked lists: 走訪終止條件不再是 cur != Null, 而是 cur != head



(2) Dummy head node: 可以去掉某些增加刪除第一個節點的特例, 幫助寫程式更方便



(3) Double linked lists: 雙向的 Circular linked lists



My Opinions
Thoughts, inspirations, and suggestions

My Notes
Important Concepts worth keeping

Today: / /

DB-
單元四 以遞迴解題

1. Symbols used in grammers: x|y means: X or y ; Xy means X followed by y
(1) identifier=<letter>|<identifier><letter>|<identifier><d-git>
(2) palindrome=empty string|<letter>|<letter><pal><letter>
(3) AnBn (legal-word)=empty string|A<legal word>B
(4) Infix <identifier>|<operator><prefix><prefix>
(5) Postfix <identifier>|<postfix><postfix><operator>

*前序式&後序式 v.s. 中序式
無括號、無優先順序 有括號、有優先順序

To see the world, things dangerous to come to, to see behind walls, to draw closer, to find each other and to feel. That is the purpose of life.
- 《The Secret Life of Walter Mitty》

My Questions
Problems & Difficulties needing exploration

2. Backtracking
(1) Eight-Queen Problem: 在8×8的棋盤上放置8個皇后，任兩個皇后都不能處於同一條橫行、縱行、斜線上
solution: 每一個皇后即做一次recursion；無答案時則退回，直到八個皆在棋盤上正確的位置
(2) A search Problem: 從起點走任意不重覆的路徑到終點
solution: 到一個不重複的中點即做一次recursion，無答案時則退回，直到走到終點

My Opinions
Thoughts, inspirations, and suggestions

開拓視野，突破萬難，看見世界，貼近彼此，感受生活，這就是生活的目的。
-《白日夢冒險王》

My Footprints

心得：

單元一：在大一的計算机概论時，我很不會寫recursion，能用while就一定用while，現在才發現原來有些題目要用recursion比較好寫。

單元二：在大一下下半學期學習Java時，我对於物件導向一知半解，既不了解為什麼不把程式都寫在main裡就好，也只会把各种修飾詞的存取範圍者隔成public，我現在不但理解其中的意義，也学到繼承、多載、覆寫等非常實用的方法。

單元三：我覺得这个單元是我最熟悉的，因為以前我們就有寫过很多相关的題目，只是因為当時沒有学过Dummy head node的技巧，所以程式寫起來十分繁複，現在学到这些新技巧，相信如果再遇到一樣的問題，程式一定能精簡不少。

單元四：我以前在看程式相关的書籍時，就有讀过定義語詞的部分，所以对於前序、中序、後序的表示法的表示甫事先換有一定的了解。另外，我覺得八皇后問題十分有趣，但同時也十分具有挑战性，希望之後我能夠成功破解。

腳蹤密碼

3 / 三聲無奈
4 / 似有隱情，等待深入關懷
5 / 無心學習，多麼可惜。
6 / 看來保留很多，加油!!!
7 / 期待你更多投入，加油!!
8 / 巴望你暢所欲言，加油!
9 / 印象深刻久久
10 / 十分感動!
10+ / 可圈可點!!!