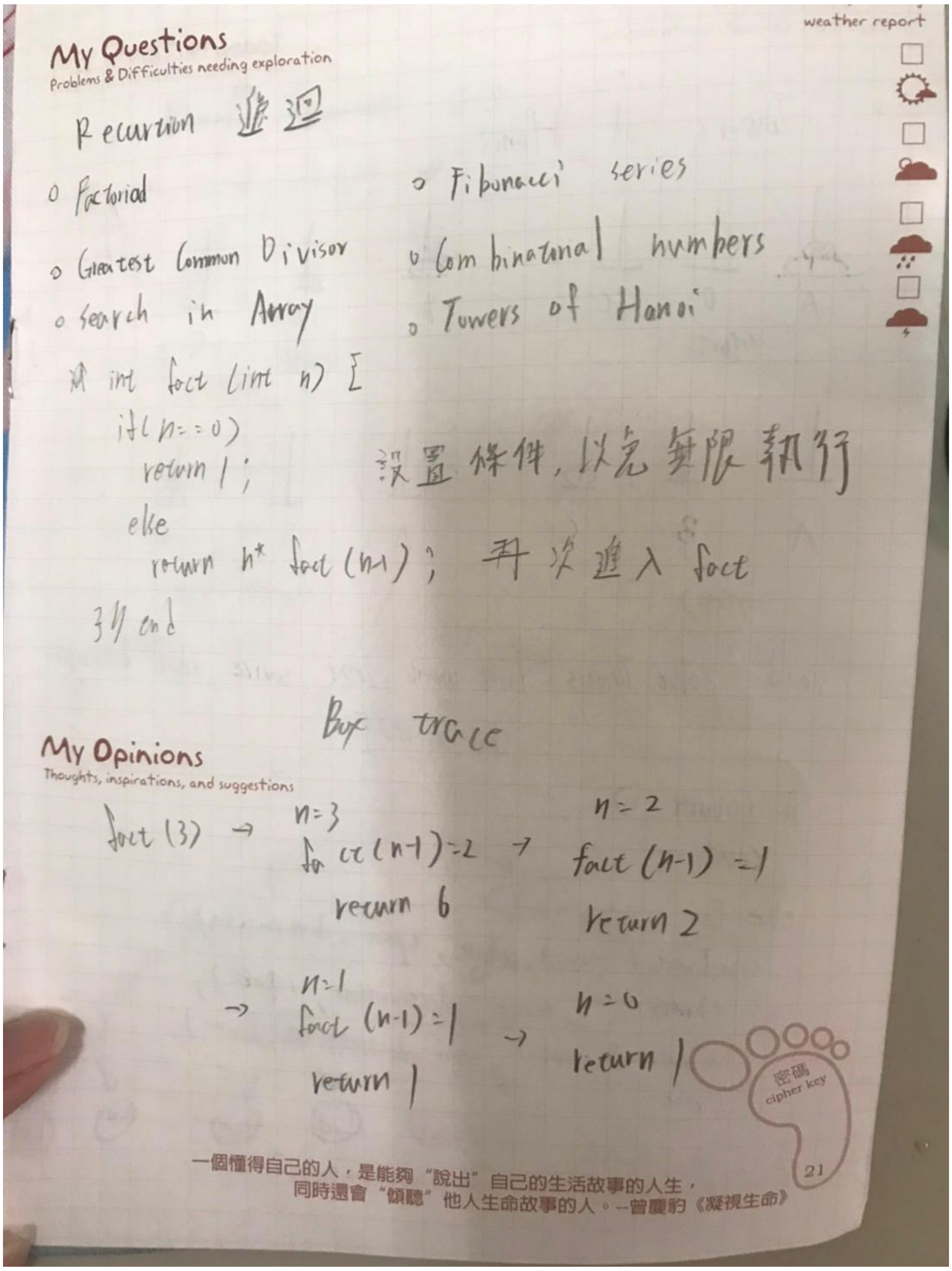
筆記

資工三甲_10724128_吳宇哲



weather report

My Questions

Problems & Difficulties needing exploration

Recurzion 遞迴

- ○ Factorial
- ○ Fibonacci series
- ○ Greatest Common Divisor
- ○ Combinatorial numbers
- ○ search in Array
- ○ Towers of Hanoi

```
例 int fact (int n) {
     if (n==0)
       return 1;          設置條件,以免無限執行
     else
       return n* fact(n-1);   再次進入fact
   }// end
```

Box trace

My Opinions

Thoughts, inspirations, and suggestions

```
fact(3) →  n=3                 →  n=2
           fact(n-1)=2             fact(n-1)=1
           return 6                return 2

        →  n=1                 →  n=0
           fact(n-1)=1             return 1
           return 1
```

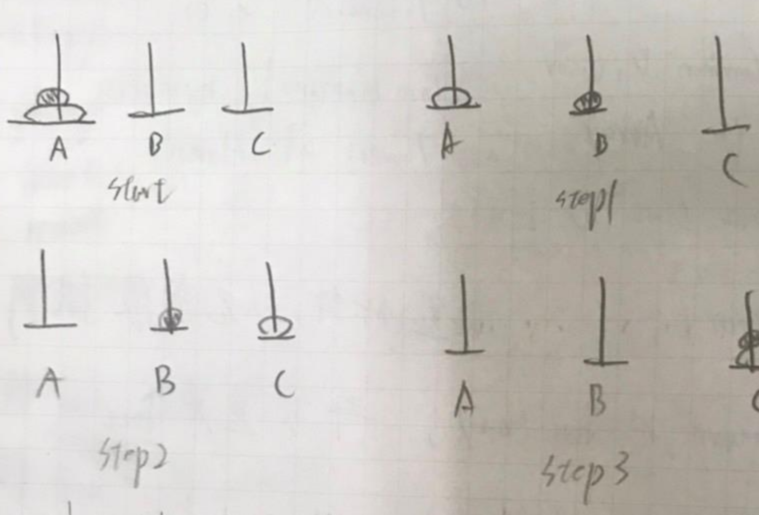
密碼 cipher key

21

一個懂得自己的人，是能夠"說出"自己的生活故事的人生，
同時還會"傾聽"他人生命故事的人。--曾慶豹《凝視生命》

Today: / /

Towers of Hanoi



```
void solve Towers (int count, char source, char destenation
                   char spare) {
  if (count = 1)
     cout ...;
  else {
    solveTowers ( count-1, source, spare, destination);
    Solve Towers (  1, source, destination, spare);
    solve Towers ( count-1, spare, destination, source);
  } // else
} // solve Towers
```

My Questions

Problems & Difficulties needing exploration

My learning weather report

Binary Recurtion

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

My Opinions

Thoughts, inspirations, and suggestions

$$\begin{tikzcd}
 & & & (3,8) \arrow[dll] \arrow[drr] & & & & \\
 & (3,4) \arrow[dl] \arrow[dr] & & & & (7,4) \arrow[dl] \arrow[dr] & & \\
(3,2) \arrow[d] \arrow[dr] & & (5,2) \arrow[d] \arrow[dr] & & (7,2) \arrow[d] \arrow[dr] & & (9,2) \arrow[d] \arrow[dr] & \\
(3,1) & (4,1) & (5,1) & (6,1) & (7,1) & (8,1) & (9,1) & (10,1)
\end{tikzcd}$$

密碼 cipher key

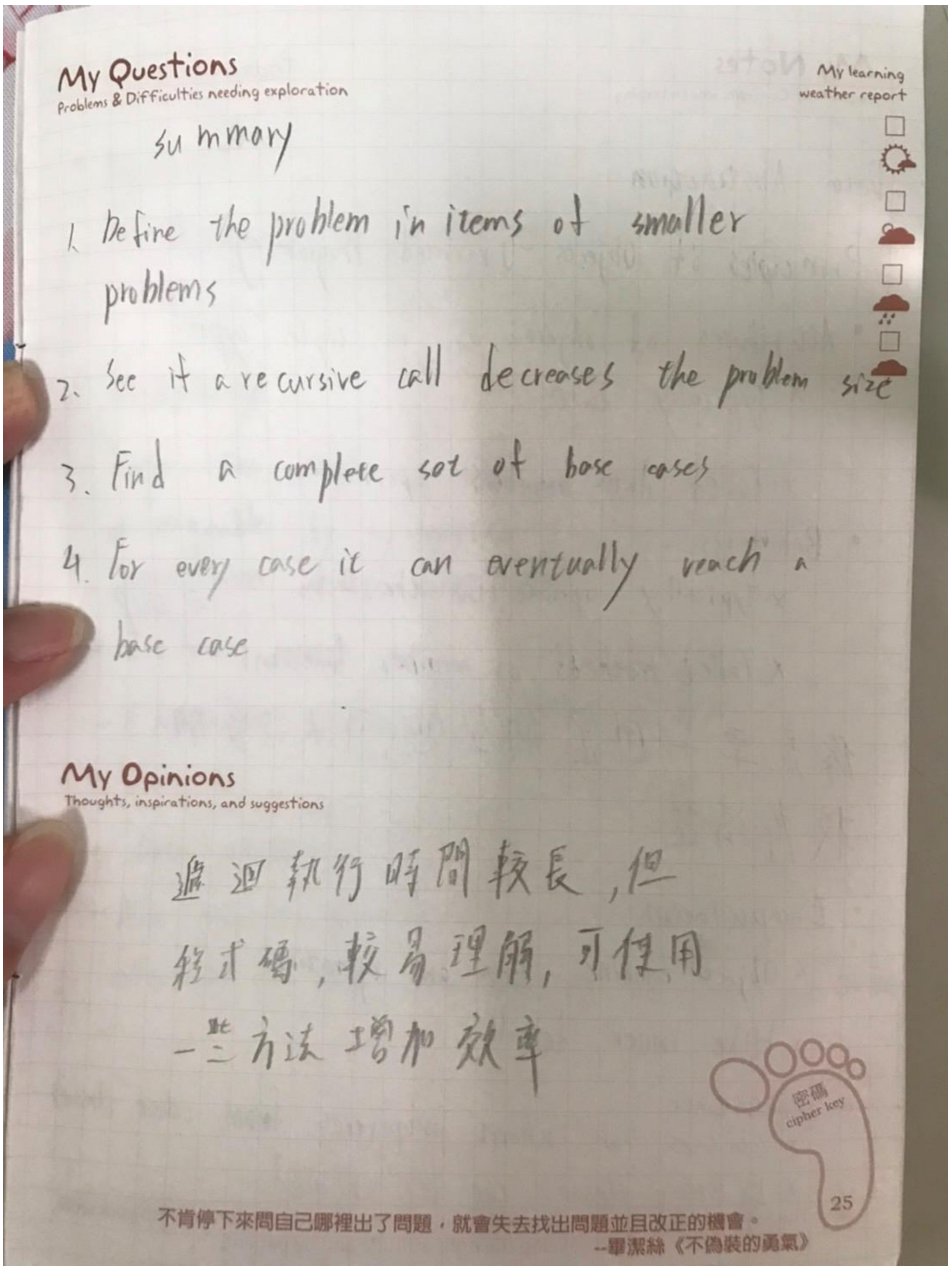
Mulitplying Rabbits (Fibonacci)

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

Box trace

n = 5	n = 4	n = 3	n = 2
rabbit (4) = 3	rabbit ( ) = 2	rabbit ( ) = 1	return 1
rabbit (3) = 2	rabbit ( ) = 1	rabbit ( ) = 1	n = 1
return 5	return 3	return 2	return 1

可用空間換時間, 以免重覆作用



My Questions

Problems & Difficulties needing exploration

My learning weather report

summary

1. Define the problem in items of smaller problems
2. See if a recursive call decreases the problem size
3. Find a complete sot of base cases
4. For every case it can eventually reach a base case

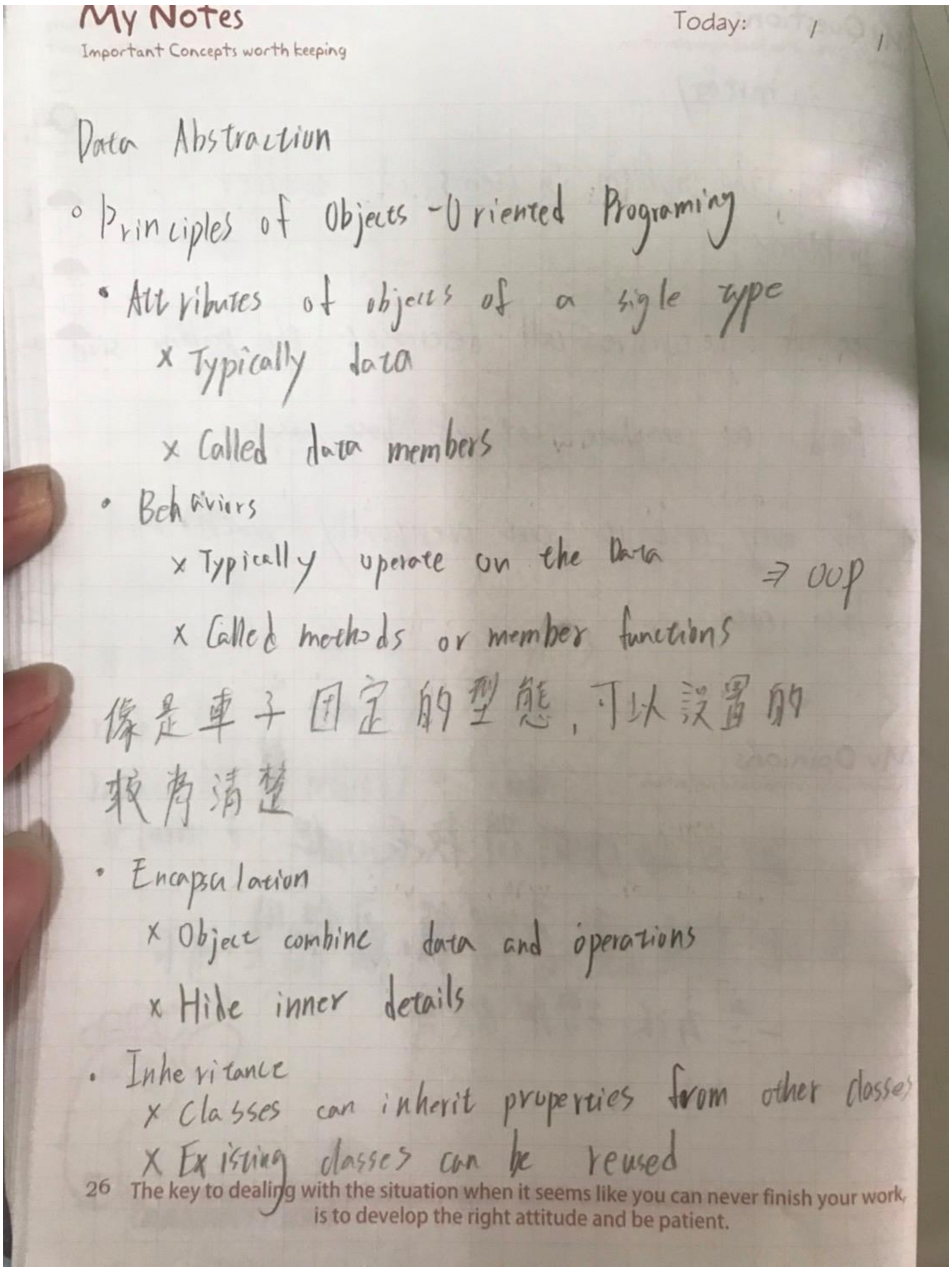
My Opinions

Thoughts, inspirations, and suggestions

遞迴執行時間較長，但程式碼，較易理解，可使用一些方法增加效率

密碼 cipher key

不肯停下來問自己哪裡出了問題，就會失去找出問題並且改正的機會。
--畢潔絲《不偽裝的勇氣》

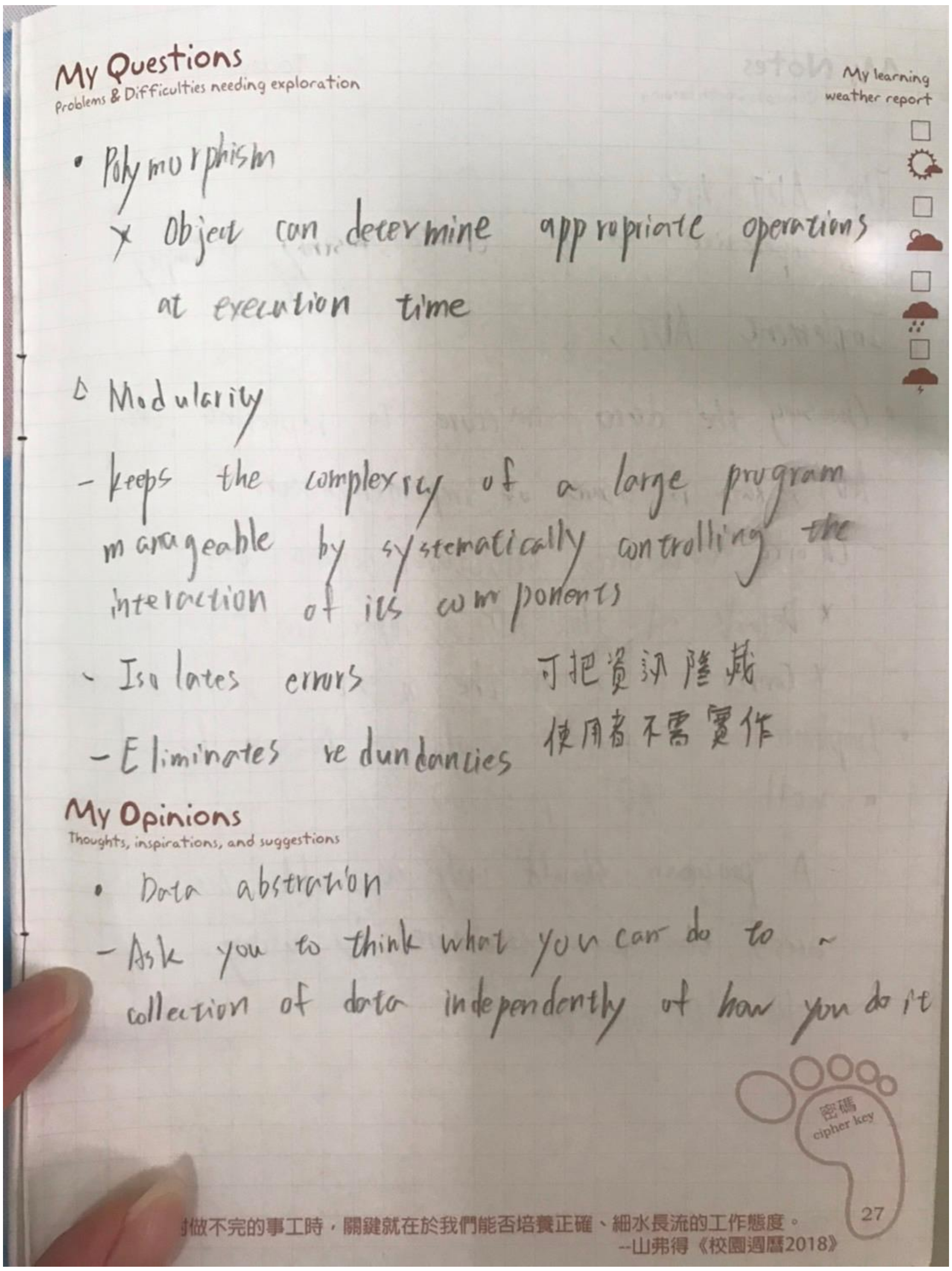


Data Abstraction

- Principles of Objects-Oriented Programing
  - Attributes of objects of a single type
    - x Typically data
    - x Called data members
  - Behaviors
    - x Typically operate on the Data ⇒ OOP
    - x Called methods or member functions

像是車子固定的型態，可以設置的較為清楚

  - Encapsulation
    - x Object combine data and operations
    - x Hide inner details
  - Inheritance
    - x Classes can inherit properties from other classes
    - x Existing classes can be reused



My Questions
Problems & Difficulties needing exploration

My learning weather report

☐ ☐ ☐ ☐

- Polymorphism
  - Object can determine appropriate operations at execution time

- Modularity
  - keeps the complexity of a large program manageable by systematically controlling the interaction of its components
  - Isolates errors 可把資訊隱藏
  - Eliminates redundancies 使用者不需實作

My Opinions
Thoughts, inspirations, and suggestions

- Data abstraction
  - Ask you to think what you can do to a collection of data independently of how you do it

密碼 cipher key

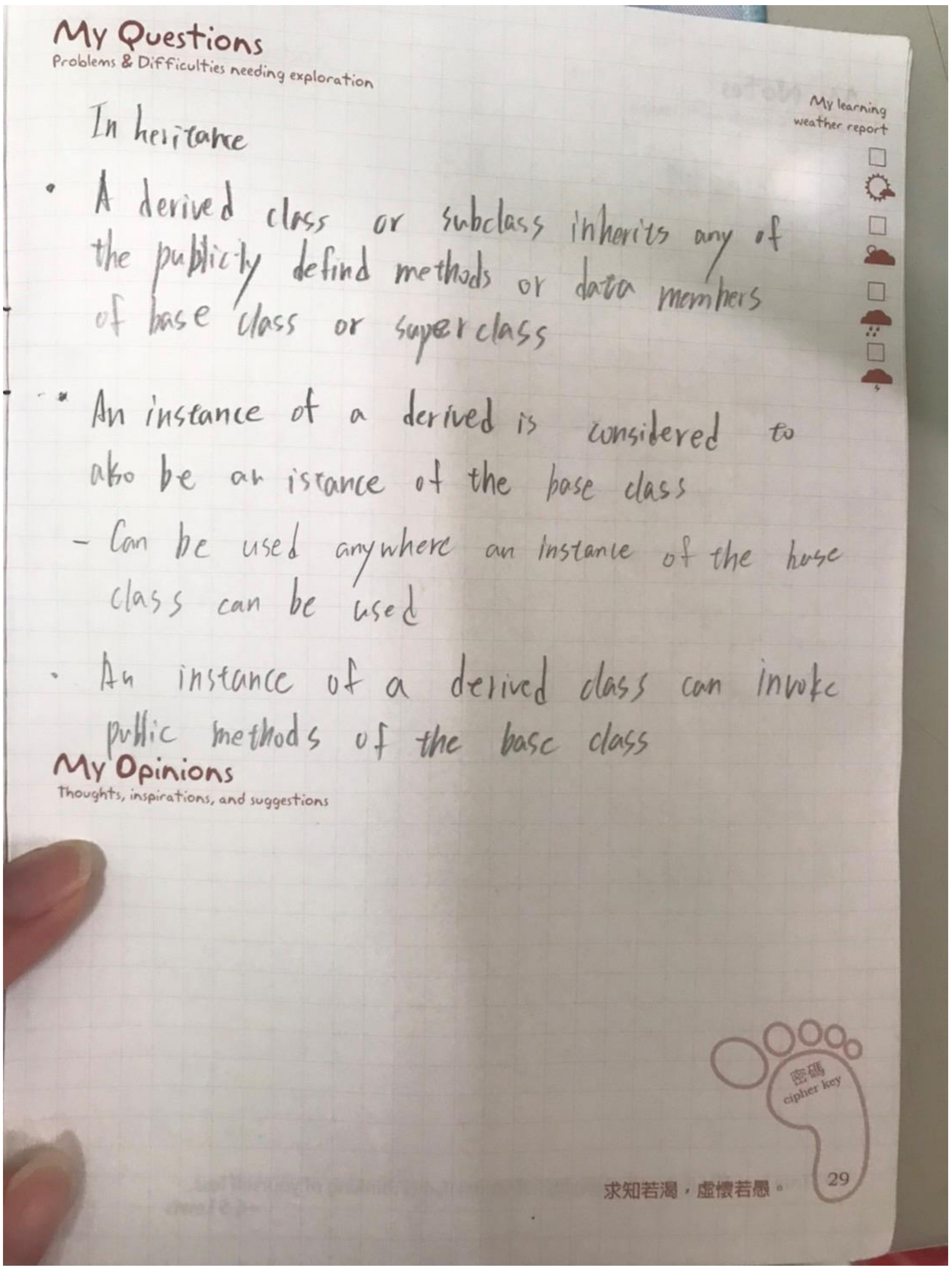
做不完的事工時，關鍵就在於我們能否培養正確、細水長流的工作態度。
—山弗得《校園週曆2018》

The ADT list

設置 operation → eg. create, destroy, is Empty ....

Implement ADT's

- Choosing the data structure to represent the ADT's data is apart of implementation
  - Choice of a data structure depend on
    - x Details of the ADT's operations
    - x Context in which the operation will be used
- Implementation details should be hidden behind a wall of ADT operations
  - A program should only be able to acess the data structure by using the ADT operation



My Questions

Problems & Difficulties needing exploration

My learning weather report

☐ ☐ ☐ ☐

In heritance

- A derived class or subclass inherits any of the publicly defind methods or data members of base class or superclass
- An instance of a derived is considered to also be an istance of the base class
  - Can be used anywhere an instance of the base class can be used
- An instance of a derived class can invoke public methods of the base class

My Opinions

Thoughts, inspirations, and suggestions

密碼 cipher key

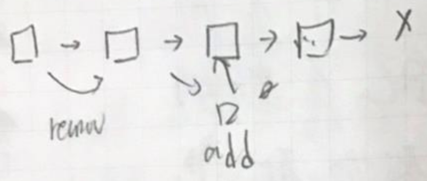
My Notes

Important Concepts worth keeping

Today: / /

Linked list

指標 = 門牌



int *p; undefined, not NULL

p=&x; &x = 房子 x 的門牌

p = new int 一棟新房

delete p; 歸還房子

p = NULL; 清空

先 delete 在 NULL，不然記憶體會掛

```
int arraysize = 50;
double * anArray = new double [arraysize];
```
動態陣列

delete [] oldArray

用 for 一間一間搬家