1. 定義：  
   遞歸方法：一個方法體內調用它自身。

２．如何理解遞歸方法？  
方法遞歸包含了一種隱式的循環，它會重複執行某段代碼，但這種重複執行無須循環控制。

遞歸一定要向已知方向遞歸，否則這種遞歸就變成了無窮遞歸，類似於死循環。

３．舉例：

// 例1：計算1-n之間所有自然數的和

**public** **int** getSum(**int** n) {// 3

**if** (n == 1) {

**return** 1;

} **else** {

**return** n + getSum(n - 1);

}

}

// 例2：計算1-n之間所有自然數的乘積:n!

**public** **int** getSum1(**int** n) {

**if** (n == 1) {

**return** 1;

} **else** {

**return** n \* getSum1(n - 1);

}

}

//例3：已知有一個數列：f(0) = 1,f(1) = 4,f(n+2)=2\*f(n+1) + f(n),

//其中n是大於0的整數，求f(10)的值。

**public** **int** f(**int** n){

**if**(n == 0){

**return** 1;

}**else** **if**(n == 1){

**return** 4;

}**else**{

**return** 2\*f(n - 1) + f(n - 2);

}

}

//例4：斐波那契數列

//例5：漢諾塔問題

//例6：快排