

# 國立麻尾高級中學

## 學生自主學習紀錄簿



學年度：108

班級：101

座號：27

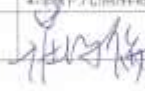


姓名：林詠智

# 目 錄

一、自主學習計畫申請書.....	1
二、自主學習紀錄.....	2
三、自主學習成果.....	12

# 一、自主學習計畫申請書

國立虎尾高中 108 學年第一學期學生自主學習計畫申請書				
申請人	林詠智	班級	101	座號 27
家長簽名	林明德		申請時數	1 節/週
學習夥伴 (無必填)	_____ 班 _____		協助專家 (無必填)	
計畫名稱	Java 基礎程式設計入門	相關學科	資訊	
內容說明	學習初階 Java 程式設計，以 Java 設計解題程式			
預計進度(週計畫)				
週次	內容			備註(場地、設備等)
1	Java 三大平台/Java SE 架構			電腦
2	Java 程式電匯下載/環境變數設定			電腦
3	使用記事本撰寫/編譯，NetBeans 的下載與使用			電腦
4	主方法 main/敘述句的使用，資料輸出(print, println, printf)			電腦
5	變數宣告:整數(byte, int) /浮點數(float, double)			電腦
6	變數宣告:字元(char)，字串(String)，布林值(boolean) 整數(short, long)			電腦
7	常數宣告(final)，運算子的分類及使用			電腦
8	參考型別，使用 Scanner 輸入			電腦
9	java 流程控制:判斷式(switch-case)，(if-else(單向判斷式，雙向判斷式))的使用			電腦
10	java 流程控制:判斷式(if-else(多向判斷式，巢狀判斷式))的使用			電腦
11	java 流程控制:迴圈結構(for 迴圈)			電腦
12	java 流程控制:迴圈結構(巢狀 for 迴圈)			電腦
13	java 流程控制:迴圈結構(while, do-while)			電腦
14 - 15	break - continue 與標籤 label 的使用			電腦
16	一維陣列，二維陣列的使用			電腦
17	製作解題程式 — 排列組合計算，判斷質數，找出陣列中的資料			電腦
18	製作解題程式 — 陣列反轉，矩陣相加，計算正 n 邊形內角和			電腦
19	製作解題程式 — 以速度總時間計算物體移動的			電腦

	加速度、平均速度、位移，找出最大公因數最小公倍數	
20	製作解題程式 — 計算比賽分數，到 6 分結束比賽，A or B or C 隊勝力	電腦
所需設備	電腦	
成果報告形式	<input type="checkbox"/> 書面報告 <input type="checkbox"/> 影音檔 <input type="checkbox"/> 動態發表會 <input type="checkbox"/> 其他(請說明): 1.製作九個解題程式	
導師	 (簽章)	指導老師  (簽章)
審查意見欄(學生勿填)		
<input checked="" type="checkbox"/> 通過 <input type="checkbox"/> 不通過 審查意見:		
委員簽章 		
備註：本申請表週次不足可自行延伸。		

## 二、自主學習紀錄

國立虎尾高級中學學生自主學習紀錄表			
紀錄人姓名	許詠賢	班級	101
填表日期	108 年 11 月 21 日 星期 四		
計畫週別	1		
執行內容 概要	<p>Java 三大平台：</p> <ul style="list-style-type: none"> <li>Java SE ← Java EE, Java ME 的基礎，提供一般程式開發。</li> <li>Java EE ← 以技術標準所組成，符合企業需要。</li> <li>Java ME ← 為機上盒、行動電話、嵌入式電子設備提供。</li> </ul> <p>Java SE 架構：</p> <ul style="list-style-type: none"> <li>JDK: Java 標準版開發工具。</li> <li>JRE: Java 執行環境</li> <li>JVM: Java 虛擬機器</li> </ul>		
自我檢核	是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：		
指導老師 意見欄			
指導老師簽章	教師張安校		

國立虎尾高級中學學生自主學習紀錄表			
紀錄人姓名	許詠賢	班級	101
填表日期	108 年 11 月 28 日 星期 五		
計畫週別	2		
執行內容 概要	<p>環境變數設定：</p> <ol style="list-style-type: none"> <li>1. 複製 JDK 安裝路徑</li> <li>2. 開啟「命令提示字元」</li> <li>輸入 set path = 安裝路徑</li> <li>3. 測試 javac 與 java。</li> </ol>		
自我檢核	是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：		
指導老師 意見欄			
指導老師簽章	教師張安校		

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	林詩智	班級	101	座號	21
填表日期	108 年 12 月 5 日 星期四				
計畫週別	3				
執行內容 概要	使用記事本撰寫、編譯 NetBeans 的下載與使用。 寫一程式在畫面輸出。 Hello World! Job				
自我檢核	是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張安綾				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	林詩智	班級	101	座號	21
填表日期	108 年 12 月 12 日 星期日				
計畫週別	4				
執行內容 概要	主方法 main 敘述句的使用，資料輸出。 寫一程式輸出 共 共共 共共共 共共共共 共共共共共				
自我檢核	是否達成預定進度： <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張安綾				

國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詩哲	班級	101	座號	27
填表日期	108 年 12 月 19 日星期四				
計畫週別	五				
執行內容 概要	變數宣告: 整數: int, byte 浮點數: float, double				
自我檢核	是否達成預定進度: <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因:				
指導老師 意見欄					
指導老師簽章	教師張宴斌				

國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詩哲	班級	101	座號	27
填表日期	108 年 12 月 26 日星期四				
計畫週別	六				
執行內容 概要	變數宣告: 字元 (char) 字串 (String) 布林值 (boolean) 整數: short, long				
自我檢核	是否達成預定進度: <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因:				
指導老師 意見欄					
指導老師簽章	教師張宴斌				



國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詠賢	班級	101	座號	17
填表日期	109年1月2日星期四				
計畫週別	7				
執行內容 概要	<p>帶狀宣告 final，運算子的分類及使用</p> <p>final 的意義為：指定數值的變數後，不可再更改。</p> <p>運算子：1. 指定運算子 (=)</p> <p>2. 算術運算子 (+, -, *, /, %)</p> <p>3. 遞增遞減運算子 (++ --)</p> <p>4. 關係運算子 (== !=, &gt;, &lt;, &lt;=, &gt;=)</p> <p>5. 邏輯運算子 (&amp;,  , !, &amp;&amp;,   )</p>				
自我檢核	<p>是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否</p> <p>未達成原因：</p>				
指導老師 意見欄					
指導老師簽章	教師張安純				

國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詠賢	班級	101	座號	17
填表日期	109年1月9日星期四				
計畫週別	8				
執行內容 概要	<p>參考型別，Scanner 輸入</p> <p>參考型別：基本型別外的資料型別，都是參考型別</p> <p>輸入：先 import java.util.Scanner</p> <p>建立 Scanner 物件 (new Scanner)</p> <p>使用 next() 取得輸入資料</p> <p>nextInt() 取得整數</p> <p>nextDouble() 取得小數..... 轉換不同型別</p>				
自我檢核	<p>是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否</p> <p>未達成原因：</p>				
指導老師 意見欄					
指導老師簽章	教師張安純				

國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詩智	班級	101	座號	27
填表日期	109年1月16日星期四				
計畫週別	9.				
執行內容 概要	<p>判斷式:</p> <pre> [switch-case] switch (變數) {   case 1:  ← break, 進入下             個 "case"             break; ← 結束 switch   default: print(y);             break; }           </pre> <p>{if-else}</p> <pre> if(判斷){   print(x); } else {   print(y); }           </pre>				
自我檢核	是否達成預定進度: <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因:				
指導老師 意見欄					
指導老師簽章	教師張安敏				

國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詩智	班級	101	座號	27
填表日期	109年2月27日星期四				
計畫週別	10.				
執行內容 概要	<p>判斷式:</p> <pre> if-else 多向判斷 if (判斷) {   print(x); } else {   print(y); }           </pre> <p>if-else</p> <pre> if (x &lt; 0) {   if (x &gt; 0) {     print(x);   } else {     print(y);   } }           </pre>				
自我檢核	是否達成預定進度: <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因:				
指導老師 意見欄					
指導老師簽章	教師張安敏				





國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詩賢	班級	101	座號	17
填表日期	109年3月19日星期四				
計畫週別	13				
執行內容 概要	<p>while, do-while 迴圈結構</p> <p>當 while 1 ↑ return 時會持續進行。</p> <p>do-while 至少執行一次，然後再判斷</p> <p>do {</p> <p>} while (1);</p> <p>↑ do-while</p> <p>while 判斷</p>				
自我檢核	是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張雲曉				

國立虎尾高級中學學生自主學習紀錄表

紀錄人姓名	林詩賢	班級	101	座號	17
填表日期	109年3月16日星期四				
計畫週別	14				
執行內容 概要	<p>break, continue 根據 while 的使用：</p> <p>break: 例如 switch 18 {</p> <pre> cases: cases: cases:   print("x")   break; }           </pre> <p>執行到 break 才結束 switch-case。 所以 X=1, X=2, X=3 都會輸出 X</p>				
自我檢核	是否達成預定進度： <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張雲曉				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	許鈺智	班級	(*)	座號	*)
填表日期	107 年 4 月 2 日 星期四				
計畫週別	5				
執行內容 概要	<p>break, continue, 標籤 label</p> <p>continue: 例如 for 迴圈</p> <pre> for (i) {     print(i);     if (判斷) continue; } </pre> <p>break: 例如 for 迴圈</p> <pre> for (i) {     for (j) {         break (break);     } } </pre> <p>break 結束外圍迴圈</p>				
自我檢核	是否達成預定進度: <input type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因:				
指導老師 意見欄					
指導老師簽章	教師張宴慈				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	許鈺智	班級	(*)	座號	*)
填表日期	107 年 4 月 9 日 星期日				
計畫週別	16				
執行內容 概要	<p>一維陣列 = 數組陣列</p> <p>宣告: int[] arr = new int[陣列長度];</p> <p>arr[0] = 1; arr[1] = 2; arr[2] = 3;</p> <p>取資料: (利用 for 迴圈)</p> <pre> for (int i = 0; i &lt; arr.length; i++) {     print(arr[i]); } </pre> <p>輸出: 1 2 3 ...</p> <p>二維陣列: 例如 int[][] arr = new int[1][1];</p> <p>arr[0][0] = 20; arr[0][1] = 30; arr[1][0] = 40; arr[1][1] = 50;</p> <p>刪除元素: 例如 arr[0][0] = 0;</p>				
自我檢核	是否達成預定進度: <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因:				
指導老師 意見欄					
指導老師簽章	教師張宴慈				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	林詠智	班級	101	座號	27
填表日期	109年4月16日 星期四				
計畫週別	17				
執行內容 概要	<p>製作解題程式：</p> <p>1. 排列組合計算 宣告 <math>n</math> 以 <math>n</math> 計算階層。</p> <p>2. 判斷質數 利用迴圈，找出函數...</p> <p>3. 找出排列中的質數 讓使用者輸入排列，使用 <code>sort()</code> 依排列後，宣告 <code>binarysearch.sort()</code> 找出找出排列中的質數。</p>				
自我檢核	是否達成預定進度： <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張雲純				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	林詠智	班級	101	座號	27
填表日期	109年4月23日 星期四				
計畫週別	18				
執行內容 概要	<p>製作解題程式：</p> <p>1. 陣列反轉 先讓使用者輸入陣列 <math>n</math>。 宣告 <math>n</math> 陣列 <code>arr</code>。 利用 <code>for</code> 迴圈，把 <math>n</math> 從 第0位輸入到 <code>arr</code> 的最後一位， 印出結果。</p> <p>2. 計算正 <math>n</math> 邊形的內角和 讓使用者輸入 <math>n</math> 次邊長， 宣告 <code>arr</code> 以 <math>n</math> 計算後輸出。</p> <p>3. 矩陣相加 宣告 <code>arr1</code> <code>arr2</code> <code>arr3</code> 讓使用者輸入 陣列，印出結果，宣告 <code>arr</code> 以 <math>n</math> 計算 執行矩陣相加後印出結果。</p>				
自我檢核	是否達成預定進度： <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張雲純				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	林詩智	班級	101	座號	27
填表日期	109 年 4 月 30 日 星期四				
計畫週別	19				
執行內容 概要	<p>製作解題程式</p> <p>1. 計算公因數、公倍數、約數、倍數 讓使用者輸入兩個 正整數，計算後 印出結果。</p> <p>2. 找出最大公因數、最小公倍數 讓使用者輸入兩個 正整數，宣告 IsGCD 方法，計算公因數和公倍數後 輸出。</p>				
自我檢核	是否達成預定進度： <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張宴珍				

國立虎尾高級中學學生自主學習紀錄表					
紀錄人姓名	林詩智	班級	101	座號	27
填表日期	109 年 5 月 7 日 星期四				
計畫週別	20				
執行內容 概要	<p>製作解題程式</p> <p>1. 計算比賽分數，到 6 日結束比賽，由 50 分開始。 讓使用者選擇加分降分，加隨機數，直到其中一隊 分數大於等於 6 分，離開 while 迴圈，印出結果。</p>				
自我檢核	是否達成預定進度： <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否 未達成原因：				
指導老師 意見欄					
指導老師簽章	教師張宴珍				

## 三、自主學習成果

### 1. 計畫名稱

Java 基礎程式設計入門

### 2. 計畫說明

學習 Java 基本語法 並使用 Java 寫九支解題程式。

### 3. 系統介紹

共有九支程式，每支程式都可以讓使用者輸入想求的值，達到使用者想要的解題功能。

包括

- I. 找出陣列中的資料位置
- II. 以速度總時間計算物體移動的加速度
- III. 平均速度、位移
- IV. 判斷質數
- V. 排列組合計算



VI. 陣列反轉

VII. 矩陣相加

VIII. 計算內角和

IX. 計算比賽分數, 到 6 分結束比賽, A or B or C  
隊勝利

X. 找出最大公因數和最小公倍數

## 4. 成果介紹

### I. 找出陣列中的資料位置

使用 For 迴圈, 讓使用者輸入陣列長度和陣列內容, 再宣告 `binarySerch_sort(int[] sortedArray, int target, int start, int end);` `sortedArray`: 排序好的陣列、`target`: 尋找的目標、`start`: 尋找的開始位置、`end`: 尋找的結束位置。

使用 `binarySearch_sort ()` 在進行之前必須進行排序, 如果不排序, 則結果是不確定的。所以利用 `sort(int[] arr)` 方法按小到大進行排序。

以下為程式碼：

```
import java.util.Arrays;
import java.util.Scanner;

public class Case_1 {
    public static int binarySerch_sort(int[] sortedArray, int target, int start, int end) {
        if (start < 0 || end >= sortedArray.length) {
            return -1;
        }
        while (start <= end) {
            int mid = (start + end) / 2;
            if (sortedArray[mid] == target) {
                return mid;
            } else if (target > sortedArray[mid]) {
                start = mid + 1;
            } else if (target < sortedArray[mid]) {
                end = mid - 1;
            }
        }
        return -1;
    }

    public static void main(String[] args) {
        //讓使用者輸入陣列長度
        Scanner arr = new Scanner(System.in);
        System.out.println("輸入陣列長度");
        int arr_length = arr.nextInt();
        int[] array = new int[arr_length];
        //輸入陣列
        for(int i= 0 ; i< array.length;i++){
            System.out.printf("輸入第%d 個數字\n",i+1);
            array[i] = arr.nextInt();
        }
        Arrays.sort(array); //由小到大排列
    }
}
```

```

for(int a:array){
    System.out.print(a+" ");
}

System.out.println("請輸入想查詢的數字");

Scanner sca = new Scanner(System.in);
int m = sca.nextInt();

//找出陣列中的位置
int v = binarySerch_sort(array,m,array.length-arr_length,array.length-1);
if (v >= 0) {
    System.out.printf(m + "的位置為:%d %n", v);
}else{
    System.out.println("找不到此數");
}

}
}

```

以下為測試畫面：

<pre> run: 輸入陣列長度 5 輸入第1個數字 8 輸入第2個數字 2 輸入第3個數字 6 輸入第4個數字 4 輸入第5個數字 3 2 3 4 6 8 請輸入想查詢的數字 4 4的位置為:2 </pre>	<pre> run: 輸入陣列長度 5 輸入第1個數字 1 輸入第2個數字 582 輸入第3個數字 93 輸入第4個數字 54 輸入第5個數字 25 1 25 54 93 582 請輸入想查詢的數字 93 93的位置為:3 </pre>	<pre> run: 輸入陣列長度 6 輸入第1個數字 2 輸入第2個數字 5 輸入第3個數字 31 輸入第4個數字 4 輸入第5個數字 5 輸入第6個數字 31 2 4 5 5 31 31 請輸入想查詢的數字 81 找不到此數 </pre>
--	--	---

## II. 以速度總時間計算物體移動的加速度、平均速度、位移

使用 Scanner 類別、switch-case 結構，讓使用者輸入時間、初速度、末速度和選擇所要計算的公式。

以下為程式碼：

```
import java.util.InputMismatchException;
import java.util.Scanner;

public class Case_2 {
    public static void main(String[] args) {

        Scanner sca = new Scanner(System.in);

        System.out.println("輸入總時間");
        int t = sca.nextInt();
        System.out.println("輸入初速");
        int Vo = sca.nextInt();
        System.out.println("輸入末速");
        int v = sca.nextInt();

        System.out.println("要取加速度請按 1; ");
        System.out.println("要取平均速度請按 2;");
        System.out.println("要取位移請按 3");
        System.out.println("全部都要請按 4");

        float a =Vo-v/ t*t;
        float b =Vo-v/ t;
        float s =(Vo*t) + (a*t*t/2) ;
```

```

try {
    int F = sca.nextInt();
    switch (F) {
        case 1:
            System.out.printf("加速度= %.2fm/s*s \n", a);
            break;
        case 2:
            System.out.printf("平均速度= %.2f m/s \n", b);
            break;
        case 3:
            System.out.printf("位移= %.2f m \n", s);
            break;
        case 4:
            System.out.printf("加速度= %.2f m/s*s, 平均速度=
                               %.2fm/s, 位移= %.2f m \n", a, b, s);
            break;
        default:
            System.out.println("輸入應為 1 到 4");
            break;
    }
} catch (InputMismatchException im) {
    System.out.println(im + "輸入應為 1 到 4!!");
}

}

```

以下為測試畫面：

輸入總時間 5 輸入初速 4 輸入末速 6 要取加速度請按1; 要取平均速度請按2; 要取位移請按3 全部都要請按4 4 加速度= -1.00 m/s*s, 平均速度= 3.00 m/s, 位移= 7.50 m	run: 輸入總時間 10 輸入初速 5 輸入末速 2 要取加速度請按1; 要取平均速度請按2; 要取位移請按3 全部都要請按4 2 平均速度= 5.00 m/s	輸入總時間 13 輸入初速 5 輸入末速 12 要取加速度請按1; 要取平均速度請按2; 要取位移請按3 全部都要請按4 3 位移= 487.50 m
--	---	--

### III.判斷質數

讓使用者輸入一數傳入 pri(), pri(int n)方法中使用 For 迴圈 把每一個小於 n 的數值拿來除以 n, 用判斷式 if 讓有因數的數 return 0; 在 main 方法中使用判斷式 if 、continue 關鍵字去除回傳值為 0 的數, 並印出 n。

以下為程式碼:

```
import java.util.Scanner;
public class Case_3 {
    //判斷質數
    public static int pri(int n) {
        for (int i = 2; i < n; i++) {
            if (n % i == 0) { //抓出有因數的數
                return 0;
            }
        }
        return n;
    }
}
```



```

public static void main(String[] args) {
    Scanner sca =new Scanner(System.in);

    System.out.print("輸入數字，計算質數:");
    int n=sca.nextInt();

    if (pri(n) == 0 || pri(n) == 1) { //排除有因數的數和 1
        System.out.println("不是質數");
    }else{
        System.out.println("是質數"); //印出質數
    }
}
}

```

以下為測試畫面：

輸入數字，計算質數:199 是質數	輸入數字，計算質數:5153 是質數	輸入數字，計算質數:513 不是質數
----------------------	-----------------------	-----------------------

## IV. 排列組合計算

創造  $f(\text{int } n)$  方法，利用遞迴計算  $n$  階層；創造  $P(\text{int } n, \text{int } k)$ 、 $C(\text{int } n, \text{int } k)$  方法，使用  $f(\text{int } n)$  計算出  $n$ 、 $k$  階層後，計算出排列數、組合數。

以下為程式碼：

```
import java.util.Scanner;

public class Case_4 {
    //計算階層
    static int f(int n) {
        if (n > 0) {
            return (n * f(n - 1));
        } else {
            return 1;
        }
    }

    //計算排列數
    static int P(int n, int k) {
        if (n < k | n == 0 | k == 0) {
            return 1;
        }
        return (f(n) / f(n - k));
    }
}
```

```

//計算組合數
static int C(int n, int k) {
    if (n < k | n == 0 | k == 0) {
        return 1;
    }
    return P(n, k) / f(k);
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    System.out.println("P 為排列數");
    System.out.println("p(5,3)=60,為 5 個元素中取 3 個的排列數量");

    System.out.println("C 為組合數,例如:C(5,3)=10");
    System.out.println("C(5,3)=10,為 5 個元素中取 3 個的組合數量");

    System.out.println("請選擇 P OR C");
    String ch = sc.next();

    boolean boo = true;
    do {

        System.out.print("請輸入\n 元素:");
        int n = sc.nextInt();
        System.out.print("取");
        int k = sc.nextInt();

        //避免數字過大
        if (n > 12 || k > 12){
            System.out.println("數字過大請重新輸入(須<12)");
            continue;
        }
    }

```

```

        boo =false;

        switch (ch) {
        case "c":
        case "C":
            System.out.println(C(n, k));
            break;
        case "p":
        case "P":
            System.out.println(P(n, k));
        }
    } while (boo); //迴圈直到 n,k 小於 12

}

}

```

以下為測試畫面：

<pre> run: P為排列數 p(5,3)=60,為5個元素中取3個的排列數量 C為組合數,例如:C(5,3)=10 C(5,3)=10,為5個元素中取3個的組合數量 請選擇P OR C c 請輸入 元素:5 取2 10 </pre>	<pre> run: P為排列數 p(5,3)=60,為5個元素中取3個的排列數量 C為組合數,例如:C(5,3)=10 C(5,3)=10,為5個元素中取3個的組合數量 請選擇P OR C C 請輸入 元素:12 取21 數字過大請重新輸入(須&lt;12) 請輸入 元素:9 取5 126 </pre>	<pre> run: P為排列數 p(5,3)=60,為5個元素中取3個的排列數量 C為組合數,例如:C(5,3)=10 C(5,3)=10,為5個元素中取3個的組合數量 請選擇P OR C P 請輸入 元素:6 取5 720 </pre>
---	---	--

## V. 陣列反轉

讓使用者輸入陣列 n 長度和陣列內容;宣告空陣列 turn 長度為 n.length，使用 For 迴圈將 n 陣列中的數值，從位置 0 開始，放進 turn 陣列的最後一位。

以下為程式碼：

```
import java.util.Scanner;

/*陣列反轉*/
public class Case_5 {
    public static void main(String[] args) {
        Scanner sca = new Scanner(System.in);

        //讓使用者輸入陣列長度
        System.out.println("輸入陣列長度");
        int arr_length = sca.nextInt();
        int[] n = new int[arr_length];
        //輸入陣列
        System.out.println("輸入陣列內容");
        for(int i= 0 ; i< n.length;i++){
            System.out.printf("輸入第%d 個數字\n",i+1);
            n[i] = sca.nextInt();
        }

        System.out.println("陣列反轉後為");
        for(int m =0;m < turn.length;m++){
            System.out.print(turn[m]+" ");
        }
        System.out.println();
    }
}
```

```

System.out.println("\n");

int[] turn=new int[n.length];

//執行陣列反轉
for(int i=0;i<n.length;i++){
    turn[i]=n[n.length -1-i];
}

//印出結果
System.out.println("陣列反轉後為");
for(int m =0;m < turn.length;m++){
    System.out.print(turn[m]+" ");
}
System.out.println();

}
}

```

以下為測試畫面：

<p>run:</p> <p>輸入陣列長度 5</p> <p>輸入陣列內容 輸入第1個數字 1 輸入第2個數字 5 輸入第3個數字 7 輸入第4個數字 8 輸入第5個數字 9</p> <p>陣列反轉後為 9 8 7 5 1</p>	<p>run:</p> <p>輸入陣列長度 6</p> <p>輸入陣列內容 輸入第1個數字 1 輸入第2個數字 2 輸入第3個數字 4 輸入第4個數字 5 輸入第5個數字 3 輸入第6個數字 1</p> <p>陣列反轉後為 1 3 5 4 2 1</p>	<p>run:</p> <p>輸入陣列長度 6</p> <p>輸入陣列內容 輸入第1個數字 2 輸入第2個數字 3 輸入第3個數字 1 輸入第4個數字 5 輸入第5個數字 41 輸入第6個數字 21</p> <p>陣列反轉後為 21 41 5 1 3 2</p>
---	---	---



## VI. 矩陣相加

宣告 `input_show_array` 方法讓使用者輸入兩個二維陣列: `arr1`、`arr2`; 宣告 `sum` 方法宣告 `arr3`, 利用 For 迴圈讓 `arr3=arr1+arr2` 達成矩陣相加, 並輸出結果。

以下為程式碼：

```
import java.util.Scanner;

public class Case_6 {
    static int m;
    static int n;

    //矩陣相加
    public static int[][] sum(int[][] arr1,int[][] arr2){
        //arr3 存放 arr1 + arr2
        int[][] arr3 =new int[m][n];

        for(int i=0;i<arr1.length;i++){
            for(int v=0;v<arr1[i].length;v++){
                arr3[i][v] = arr1[i][v]+arr2[i][v];
            }
        }

        return arr3;
    }
}
```

//讓使用者輸入陣列 顯示輸入結果

```
public static int[][] input_show_array(int[][] arr,int n){
    Scanner sca =new Scanner(System.in);
    System.out.printf("輸入陣列%d 內容\n",n);
    for(int i=0;i<arr.length;i++){
        for(int v=0;v<arr[i].length;v++){
            System.out.printf("輸入第%d 排 第%d 個: ",i+1,v+1);
            arr[i][v]=sca.nextInt();
        }
    }
    System.out.printf("陣列%d 內容為:",n);
    for(int column[] : arr){
        System.out.println();
        for(int row : column){
            System.out.print(" "+row);
        }
    }
    System.out.println();

    return arr;
}
```

```
public static void main(String[] args){
    Scanner sca =new Scanner(System.in);
    System.out.println("請輸入陣列大小");
    System.out.print("輸入{行}:");
    m =sca.nextInt();
    System.out.print("輸入{列}:");
    n =sca.nextInt();

    int[][] arr1 =new int[m][n];
    int[][] arr2 =new int[m][n];

    System.out.println();
    //讓使用者輸入陣列 1 顯示輸入結果
    arr1=input_show_array(arr1,1);
    System.out.println();
}
```

```
//讓使用者輸入陣列 2 顯示輸入結果
arr2=input_show_array(arr2,2);
System.out.println();

int[][] new_arr = sum(arr1,arr2);

//印出相加結果
System.out.print("矩陣相加結果為:");
for(int column[] : new_arr){
    System.out.println();
    for(int row : column){
        System.out.print(" "+row);
    }
}
System.out.println();

}
```

以下為測試畫面：

<p>run:</p> <p>請輸入陣列大小</p> <p>輸入(行):2</p> <p>輸入(列):3</p> <p>輸入陣列1內容</p> <p>輸入第1排 第1個: 1</p> <p>輸入第1排 第2個: 2</p> <p>輸入第1排 第3個: 3</p> <p>輸入第2排 第1個: 4</p> <p>輸入第2排 第2個: 5</p> <p>輸入第2排 第3個: 6</p> <p>陣列1內容為:</p> <p>1 2 3</p> <p>4 5 6</p> <p>輸入陣列2內容</p> <p>輸入第1排 第1個: 7</p> <p>輸入第1排 第2個: 8</p> <p>輸入第1排 第3個: 9</p> <p>輸入第2排 第1個: 10</p> <p>輸入第2排 第2個: 11</p> <p>輸入第2排 第3個: 12</p> <p>陣列2內容為:</p> <p>7 8 9</p> <p>10 11 12</p> <p>矩陣相加結果為:</p> <p>8 10 12</p> <p>14 16 18</p>	<p>run:</p> <p>請輸入陣列大小</p> <p>輸入(行):4</p> <p>輸入(列):2</p> <p>輸入陣列1內容</p> <p>輸入第1排 第1個: 4</p> <p>輸入第1排 第2個: 5</p> <p>輸入第2排 第1個: 2</p> <p>輸入第2排 第2個: 12</p> <p>輸入第3排 第1個: 31</p> <p>輸入第3排 第2個: 22</p> <p>輸入第4排 第1個: 5</p> <p>輸入第4排 第2個: 18</p> <p>陣列1內容為:</p> <p>4 5</p> <p>2 12</p> <p>31 22</p> <p>5 18</p> <p>輸入陣列2內容</p> <p>輸入第1排 第1個: 1</p> <p>輸入第1排 第2個: 51</p> <p>輸入第2排 第1個: 31</p> <p>輸入第2排 第2個: 51</p> <p>輸入第3排 第1個: 251</p> <p>輸入第3排 第2個: 21</p> <p>輸入第4排 第1個: 61</p> <p>輸入第4排 第2個: 34</p> <p>陣列2內容為:</p> <p>1 51</p> <p>31 51</p> <p>251 21</p> <p>61 34</p> <p>矩陣相加結果為:</p> <p>5 56</p> <p>33 63</p> <p>282 43</p> <p>66 52</p>	<p>輸入(行):5</p> <p>輸入(列):2</p> <p>輸入陣列1內容</p> <p>輸入第1排 第1個: 1</p> <p>輸入第1排 第2個: 5</p> <p>輸入第2排 第1個: 2</p> <p>輸入第2排 第2個: 34</p> <p>輸入第3排 第1個: 61</p> <p>輸入第3排 第2個: 21</p> <p>輸入第4排 第1個: 61</p> <p>輸入第4排 第2個: 31</p> <p>輸入第5排 第1個: 11</p> <p>輸入第5排 第2個: 58</p> <p>陣列1內容為:</p> <p>1 5</p> <p>2 34</p> <p>61 21</p> <p>61 31</p> <p>11 58</p> <p>輸入陣列2內容</p> <p>輸入第1排 第1個: 74</p> <p>輸入第1排 第2個: 56</p> <p>輸入第2排 第1個: 51</p> <p>輸入第2排 第2個: 23</p> <p>輸入第3排 第1個: 54</p> <p>輸入第3排 第2個: 82</p> <p>輸入第4排 第1個: 39</p> <p>輸入第4排 第2個: 12</p> <p>輸入第5排 第1個: 25</p> <p>輸入第5排 第2個: 62</p> <p>陣列2內容為:</p> <p>74 56</p> <p>51 23</p> <p>54 82</p> <p>39 12</p> <p>25 62</p> <p>矩陣相加結果為:</p> <p>75 61</p> <p>53 57</p> <p>115 103</p> <p>100 43</p> <p>36 120</p>
---	--	--

## VII. 計算正 n 邊形內角合

讓使用者輸入多邊形邊數; 宣告 `ang(int n)` 方法, 計算宣告 `out` 變數計算外角後 宣告 `in` 變數 利用互補關係 $(180-k)$ 計算出的內角, 為節省記憶體重複使用 `out` 變數傳入 `in*n`; `return out` 後印出結果。

以下為程式碼：

```
import java.util.*;
public class Case_7 {
    public static int ang(int n) {
        int out = 360 / n;
        int in = 180 - out;
        out = n * in;
        return out;
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("計算正 n 邊形內角和");
        System.out.print("請輸入邊長:");
        int n = sc.nextInt();

        if (n < 3) { //是否可以組成多邊形
            System.out.println("不成立");
        } else {
            //輸出結果
            System.out.println("內角合為"+ang(n));
        }
    }
}
```

以下為測試畫面：

<pre>run: 計算正n邊形內角和 請輸入邊長:8 內角合為1080</pre>	<pre>run: 計算正n邊形內角和 請輸入邊長:7 內角合為900</pre>	<pre>run: 計算正n邊形內角和 請輸入邊長:5 內角合為540</pre>
--	---	---

## VIII. 計算比賽分數，到 6 分結束比賽，A or B or C 隊勝力

宣告 p1、p2、p3 為 A、B、C 分數，讓使用者輸入要加分的隊伍，利用 Random 類別隨機加-3 到 3 分，使用 do-While 迴圈重複執行，直到 p1 或 p2 或 p3  $\geq 6$ ，離開迴圈輸出 X 隊勝利。

以下為程式碼：

```
import java.util.*;

/*計算比賽分數，到 5 分結束比賽，A or B 隊勝利*/
public class Case_8 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Random ran = new Random();
        int p1 = 0;
        int p2 = 0;
        int p3 = 0;
```



```

do {
    System.out.printf("目前 A:%d 分，B:%d 分，C:%d 分\n",p1,p2,p3);
    System.out.println("請輸入要加分隊伍 A or B or C");
    String team = sc.next();
    int add;

    //加隨機亂數-3-3
    if(ran.nextBoolean()){
        add=ran.nextInt(3);
    }else{
        add=ran.nextInt(3)*-1;
    }
    switch (team) {
        case "A":    //選 A
        case "a":
            p1 = p1 + add;
            break;
        case "B":    //選 B
        case "b":
            p2 = p2 + add;
            break;
        case "C":    //選 C
        case "c":
            p3=p3+add;
            break;
        default:
            System.out.println("隊伍不存在");
    }
    if (p1 >= 6 || p2 >= 6 || p3>=6) {    //如果 p1 p2 p3>6，離開迴圈
        break;
    }
} while (true);
System.out.printf("目前 A:%d 分，B:%d 分，C:%d 分\n",p1,p2,p3);

```

```

        if (p1 > p2 && p1>p3) {
            System.out.println("比賽結束 A 隊勝利");
        } else if(p2>p3 &&p2>p1){
            System.out.println("比賽結束 B 隊勝利");
        }else if(p3>p1 || p3>p2){
            System.out.println("比賽結束 C 隊勝利");

        }
    }
}

```

以下為測試畫面：

<pre> run: 目前A:0分，B:0分，C:0分 請輸入要加分隊伍 A or B or C a 目前A:2分，B:0分，C:0分 請輸入要加分隊伍 A or B or C a 目前A:4分，B:0分，C:0分 請輸入要加分隊伍 A or B or C a 目前A:6分，B:0分，C:0分 比賽結束A隊勝利 </pre>	<pre> 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:1分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:2分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:0分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:2分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:3分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:2分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:1分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:1分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:2分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:4分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:4分 請輸入要加分隊伍 A or B or C c 目前A:-4分，B:-4分，C:6分 比賽結束C隊勝利 </pre>	<pre> run: 目前A:0分，B:0分，C:0分 請輸入要加分隊伍 A or B or C a 目前A:-1分，B:0分，C:0分 請輸入要加分隊伍 A or B or C b 目前A:-1分，B:2分，C:0分 請輸入要加分隊伍 A or B or C b 目前A:-1分，B:2分，C:0分 請輸入要加分隊伍 A or B or C b 目前A:-1分，B:4分，C:0分 請輸入要加分隊伍 A or B or C b 目前A:-1分，B:6分，C:0分 比賽結束B隊勝利 </pre>
---	---	--

## IX.找出最大公因數和最小公倍數

讓使用者輸入兩數  $m$  和  $n$  後，宣告 `Fac(int n, int m)` 方法，利用巢狀 `For` 迴圈，把小於參數  $n$  的數全部拿來除  $n$ ，參數  $m$  比照辦理，利用 `if` 判斷整除  $n$ 、 $m$  的數是否相等，如是，為公因數；宣告全域變數  $s$  更新公因數，直到  $s$  為最大公因數，`return s`。

宣告整數 `fac=Fac(n,m)` 讓  $n$ 、 $m$  相乘後除以 `fac`，即為最小公倍數，直接計算  $n*m/fac$ ，並輸出最大公因數、最小公倍數。

以下為程式碼：

```
import java.util.*;
public class Case_9 {

    static int s;
    //取公因數
    public static int Fac(int n,int m){
        //把數值一個接一個抓來除 n 和 m
        for(int i=1;i<=n;i++){
            for(int v=1;v<=m;v++){
                //確定整除 n 和 m 並確定公因數
                if(n%i==0 && m%v==0 && v==i){
                    //如果直接 return 結果為 1 所以使用 static 變數儲存公因數
                    s=i;
                }
            }
        }
    }
}
```

```

public static void main(String[] args) {
    Scanner sca =new Scanner(System.in);

    System.out.println("請輸入兩數,取最大公因數和最小公倍數");

    int n=sca.nextInt();
    int m=sca.nextInt();
    int fac=Fac(n,m);

    System.out.printf("最大公因數:%d\n 最小公倍數:%d\n", fac ,n*m/fac);

}
}

```

以下為測試畫面：

<pre> run: 請輸入兩數,取最大公因數和最小公倍數 165 231 最大公因數:33 最小公倍數:1155 </pre>	<pre> run: 請輸入兩數,取最大公因數和最小公倍數 49 21 最大公因數:7 最小公倍數:147 </pre>	<pre> run: 請輸入兩數,取最大公因數和最小公倍數 234 104 最大公因數:26 最小公倍數:936 </pre>
--	--	---

## 5. 結論與心得

很幸運可以成為第一個體驗到自主學習的一屆，多虧有這個課程，讓我從寫計畫到執行，學習到如何學習自己想學的東西，雖然目前只學習到 Java 的基礎語法；在 Java 領域中的精華部份都還沒摸上，但是我的自主學習計畫還是讓我這兩個學期收穫良多。