

# 作業二

## 程式說明

### 程式要求:

- 請設計程式，將讀取陣列的內容，並回傳內容加二後的陣列。
- 在主程式(main)裡面完成：
  - 印出「輸入陣列大小：」
  - 輸入陣列大小
  - 印出「輸入第X位：」(X為輸入進來的位置)
  - 讀入陣列
  - 印出輸入的陣列
  - 印出加二後的回傳的陣列
- 在Function(PlusTwo)完成加二的動作。

### 固定函式:

```
● ● ●  
1 public static int[] PlusTwo(int[] digits) {}
```

# 設計說明

- 程式開始時，先接收使用者輸入的陣列大小
- 先宣告比輸入大小多一位的陣列用在可能發生的進位上
- 接收使用者輸入陣列內容
- 在輸出原始陣列時先切割掉宣告時多出來的一位
- 將陣列傳入 PlusTwo
- 在 PlusTwo 中，從最低位開始往最高位做迭代，判斷數字總和是否需要進位並做出相對應處理。
- 最後判斷是否需要用到陣列中多宣告的一位，回傳加二過的陣列或切割掉多餘一位的陣列。

# 自訂變數及函式

## 主程式(main):

```
● ● ●  
1 int size = scanner.nextInt();
```

▲ 用來儲存原始陣列大小

```
● ● ●  
1 int nums[] = new int[size + 1];
```

▲ 用來儲存陣列(多宣告一位)

```
● ● ●  
1 int temp_nums[] = Arrays.copyOfRange(nums, 1, size + 1);
```

▲ 儲存切割完的原始陣列

## 函式(PlusTwo):

```
● ● ●  
1 int carry = 0;
```

▲ 判斷是否需要進位

```
● ● ●  
1 int sum = 0;
```

▲ 用來計算每一位數字的總和

# 程式分析

## 1. 輔助函式 - 陣列加二



```
1 // flag 用來判斷是否需要進位
2 int carry = 0;
3 int sum = 0;
```

▲ 先宣告需要用到的變數



```
1 // 從最低位開始往回做進位
2 for (int i = digits.Length - 1; i > 0; i--) {
3     // 計算每一次總和
4     sum = digits[i] + carry;
5
6     // 假如為最低位，執行加二
7     if (i == digits.Length - 1) {
8         sum += 2;
9     }
10
11    // 假如總和為二位數即進位並改變 flag
12    if (sum > 9) {
13        digits[i] = sum - 10;
14        carry = 1;
15    } else {
16        digits[i] = sum;
17        carry = 0;
18        break;
19    }
20 }
```

1. 由最低位往回迭代
2. 每次計算該位總和
3. 假如為最低位就加二
4. 其他位假如需要進位就改變 carry，若無需進位  
則退出迴圈

# 1. 輔助函式 - 陣列加二

```
● ● ●  
1 // 假如最後 flag 為真代表需要進位，改變最高位並回傳。 反之回傳不含最高位的陣列  
2 if (carry == 1) {  
3     digits[0] = 1;  
4     return digits;  
5 } else {  
6     return Arrays.copyOfRange(digits, 1, digits.Length);  
7 }
```

1. 檢查最後是否需要進位
2. 如果需要進位就使用一開始多宣告的位置並回傳
3. 如不需要就切割掉多餘的位置並回傳

## 2. 主函式

```
● ● ●  
1 // 輸入陣列大小  
2 Scanner scanner = new Scanner(System.in);  
3 System.out.println("請輸入陣列大小:");  
4 int size = scanner.nextInt();
```

### ▲ 接收使用者輸入陣列大小

```
● ● ●  
1 // 預設多一位給可能發生的進位  
2 int nums[] = new int[size + 1];
```

### ▲ 宣告多一位大小的陣列用於可能發生的進位

```
● ● ●  
1 // 輸入陣列內容  
2 for (int i = 1; i < nums.Length; i++) {  
3     System.out.printf("輸入第%d位 : ", i);  
4     nums[i] = scanner.nextInt();  
5 }
```

### ▲ 接收使用者輸入陣列內容並存入陣列

```
● ● ●  
1 // 之後不須輸入，關閉 Scanner  
2 scanner.close();
```

### ▲ 關閉 Scanner

## 2. 主函式



```
1 // 暫時先不考慮為進位保留的位置  
2 int temp_nums[] = Arrays.copyOfRange(nums, 1, size + 1);  
3 // 印出輸入的陣列  
4 System.out.println("輸入的陣列 : " + Arrays.toString(temp_nums));
```

▲ 切割掉多餘的位置並輸出原始陣列內容



```
1 // 印出加工後的陣列 (PlusTwo)  
2 System.out.println("加二後的陣列 : " + Arrays.toString(PlusTwo(nums)));
```

▲ 輸出呼叫 PlusTwo 後的結果

# 執行結果

輸入陣列大小：

3

輸入第1位 : 9

輸入第2位 : 9

輸入第3位 : 9

輸入的陣列 : [9, 9, 9]

加二後的陣列 : [1, 0, 0, 1]

輸入陣列大小：

2

輸入第1位 : 9

輸入第2位 : 8

輸入的陣列 : [9, 8]

加二後的陣列 : [1, 0, 0]

輸入陣列大小：

1

輸入第1位 : 8

輸入的陣列 : [8]

加二後的陣列 : [1, 0]

輸入陣列大小：

3

輸入第1位 : 1

輸入第2位 : 3

輸入第3位 : 7

輸入的陣列 : [1, 3, 7]

加二後的陣列 : [1, 3, 9]