

tags: 演算法

程式作業2_b報告

資工二A 108502516 謝程偉

作法

- 使用二維vector陣列，會先用binary search判斷數列中是否有該數字存在，然後才進行merge sort，先將要插入數字存在新陣列，然後從第一行開始往下merge，每次merge結果放在新陣列中，當偵測到此行為空時就替換成新陣列。

```
C++ HW2_b
1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10
[0][0]
[1][1]
[0][0]
[2][3]
[0][0]
[1][1]
[0][0]
[3][7]
[0][0]
[1][1]
請按任意鍵繼續 . . .
```

執行結果

分析

- 先進行binary search，現存每行數列進行binary search，時間複雜度是 $O(\log(n))$ ，沒找到就進行merge sort，時間複雜度是 $O(n)$ ，由於從第一層開始merge，使用時間是 $\frac{1}{2^h}n$ (h 是目前深度，從0開始計算)，新創建用來存新數字的陣列也是 $\frac{1}{2^h}n$ ，merge的時間是 $2\frac{1}{2^h}n$ ，之後每一層都如此計算，使用時間是 $2(\frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^h})n < 2n$ ，因此是 $O(n)$ 。
- c++ map使用了紅黑樹資料結構，特色是比起一般的二元樹，它會平衡兩邊的高度，不會出現歪斜的狀況，讓search的時間複雜度一定是 $O(\log(n))$ ，而插入會透過旋轉或是變色來調整結構，時間複雜度也是 $O(\log(n))$ 較使用merge sort方法快速。