

REPORT

I. Environment

Mac OS, clang-1316.0.21.2, Xcode

II. Results

- method or solutions
 - 先判斷第一個跟最後一個硬幣是不是假的
 - 再來判斷輸入的硬幣數量與3的關係
 - (a) 3的倍數 -> 直接進行下一個步驟
 - (b) 3的倍數+1 -> 直接刪除最後一個數字
 - (c) 3的倍數+2 -> 判斷倒數第二個數字是不是假的硬幣
是的話直接輸出，不是的話就刪除最後兩個數字
 - 把剩下的數字切成3份，計算每份的和，
挑出不同的那份繼續進行計算，重新分成三小份再進行判斷
 - 當一份數字的首尾數字只差1的時候，和前後兩個數字相比，判斷哪個才是假的硬幣
 - 輸出假硬幣的排序
- time complexity of using scale: $\Theta(n)$

```
for(int i=x; i<k/3+1; i++){
    sum1 = sum1 + arr[i];
}
for(int i=k/3+1; i<2*k/3+1; i++){
    sum2 = sum2 + arr[i];
}
for(int i=2*k/3+1; i<k+1; i++){
    sum3 = sum3 + arr[i];
}
```

從73~81行的迴圈判斷

Sum1, sum2, sum3 每個拆開來單獨都需要 $n/3$ 的時間，又因為總共有3組sum，所以將 $n/3 * 3 = n$ ，得到最後的時間複雜度為 n 。