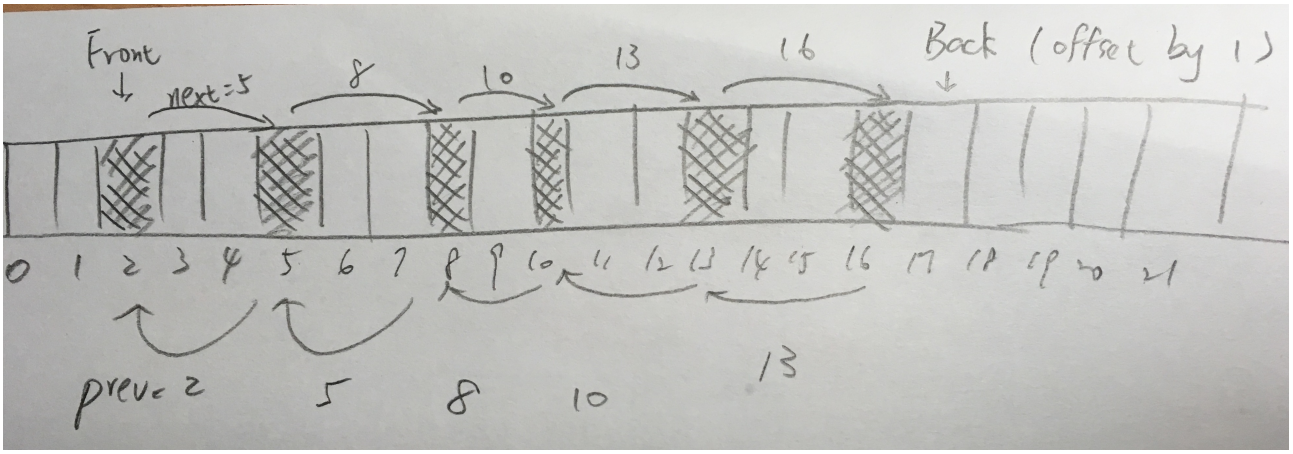


在problem2中 我使用 array 來實踐 Active List
(原本用LinkList發現速度太慢, 受不了所以改個方法寫)

雖然說資料結構屬於array, 不過背後精神卻是 Link List
這種實踐法有其特殊性與利弊，且針對此問題較 單純LinkList 來解題會更快
特殊性:

1.因為這個題目的端點一定是一左一右，所以Active list array可以分配固定大小為端點數/2
2.每一個端點都只會對應到一個Active list array中的 key 值
3.Front ,Back 指標在array 中只會增大，不會變小

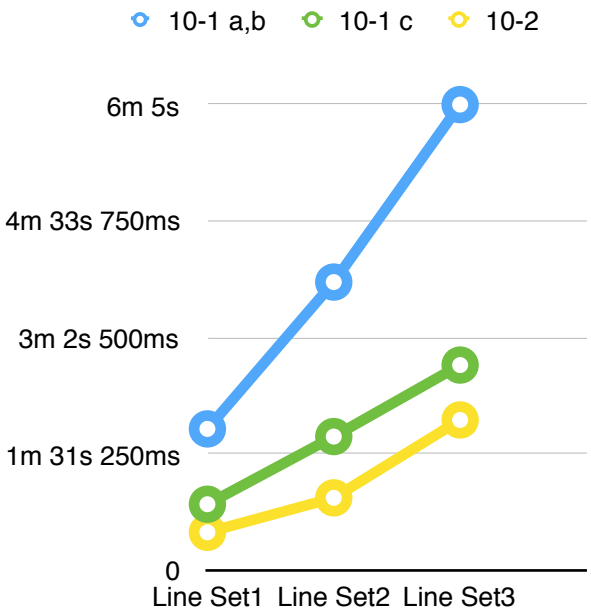
優點：

1.一個Active List 只會分配一次空間，overhead 較 LinkList小(假設LinkList分配空間的策略是一次只allocate 一個)
2.不需要指標dereference 操作, 感覺好像比較快
3.邏輯較簡單，較LinkList好寫

缺點:

1.佔用空間量較大

速度比較 : Adding time



	10-1 a , b	10-1 c	10-2
Line Set 1	111 sec	52 sec	30 sec
Line Set 2	226 sec	105 sec	57 sec
Line Set 3	365 sec	161 sec	118 sec

10-2 使用 題目內測資 運行出來的ActiveList 樣子

```
$ ./0411276_P10_2 input1.txt

front =     0, back =     0, total_active =    0
id :     0, point :     0, h=     0, next=     1, prev =     0, active = 0  Front
id :     0, point :     0, h=     1, next=     2, prev =     0, active = 0
id :     0, point :     0, h=     2, next=     3, prev =     1, active = 0
id :     0, point :     0, h=     3, next=     4, prev =     2, active = 0
processing point :000000

front =     0, back =     1, total_active =    1
id :     1, point :     5, h=     0, next=     1, prev =     0, active = 1  Front
id :     0, point :     0, h=     1, next=     2, prev =     0, active = 0  Back
id :     0, point :     0, h=     2, next=     3, prev =     1, active = 0
id :     0, point :     0, h=     3, next=     4, prev =     2, active = 0

front =     0, back =     2, total_active =    2
id :     1, point :     5, h=     0, next=     1, prev =     0, active = 1  Front
id :     0, point :    75, h=     1, next=     2, prev =     0, active = 1
id :     0, point :     0, h=     2, next=     3, prev =     1, active = 0  Back
id :     0, point :     0, h=     3, next=     4, prev =     2, active = 0

front =     0, back =     3, total_active =    3
id :     1, point :     5, h=     0, next=     1, prev =     0, active = 1  Front
id :     0, point :    75, h=     1, next=     2, prev =     0, active = 1
id :     2, point :   100, h=     2, next=     3, prev =     1, active = 1
id :     0, point :     0, h=     3, next=     4, prev =     2, active = 0  Back

front =     1, back =     3, total_active =    2
id :     1, point :     5, h=     0, next=     1, prev =     0, active = 0
id :     0, point :    75, h=     1, next=     2, prev =     0, active = 1  Front
id :     2, point :   100, h=     2, next=     3, prev =     1, active = 1
id :     0, point :     0, h=     3, next=     4, prev =     2, active = 0  Back

front =     1, back =     3, total_active =    1
id :     1, point :     5, h=     0, next=     1, prev =     0, active = 0
id :     0, point :    75, h=     1, next=     3, prev =     0, active = 1  Front
id :     2, point :   100, h=     2, next=     3, prev =     1, active = 0
id :     0, point :     0, h=     3, next=     4, prev =     1, active = 0  Back
```