



Insert at Index

ให้เขียนโปรแกรมเพื่อทำการสร้างและแก้ไข Linked List ที่มี Node Structure ดังนี้

```
typedef struct node {
    int value;
    struct node *next;
} node_t;
```

โดยจะต้องทำ operation ตาม code ที่ได้จาก input ดังนี้

i = insert at index

- แทรก (insert) Node ใหม่ลงใน Linked List ตามตาม *index* ที่กำหนด โดยให้ *node* แรก (ตำแหน่ง *head*) มี *index* เป็น 0
- หาก *index* มีค่าน้อยกว่า 0 หรือ มากกว่าความยาว List ให้แสดงผล [*<index>*: *<value>* not inserted]

d = delete at index

- ลบ Node จาก Linked List ที่ตำแหน่ง *index* และแสดงผลการลบ
 - หากลบสำเร็จจะแสดงผลในรูปแบบ [*<index>*: *<value>* deleted]
 - หากลบไม่สำเร็จจะแสดงผลในรูปแบบ [*<index>*: not deleted]

v = value at index

- แสดงค่า value จาก Node ที่ตำแหน่ง *index* ในรูปแบบ *index: value*
- แสดงผล [*<index>*: invalid index] ถ้า *index* ไม่อยู่ในตำแหน่งที่แสดงผลได้

p = print list

- แสดงค่าทั้งหมดใน List
- แสดงบรรทัดว่างถ้าเป็น List ว่าง

Input

- จำนวนแรก คือ n แทนจำนวน operation ทั้งหมด
- หลังจากนั้น n บรรทัด จะขึ้นต้นด้วยอักษรคำสั่ง i (Insert), d (Delete) v (Value) และ p (Print List) แล้วตามด้วย *index* และ ค่าที่ต้องการ Insert (เฉพาะในกรณี i)

Input

```
10
i 0 17
i 1 100
d 0
p
i 1 21
d 8
i 9 32
v 1
v 25
p
```

Output

- แสดง output จากการ Delete (d) Insert (i) Value (v) หรือ Print (p) ตามที่ระบุด้วยชุดคำสั่งใน Input

output

```
[0: 17 deleted]
100
[8: not deleted]
[9: 32 not inserted]
1: 21
[25: invalid index]
100 21
```