

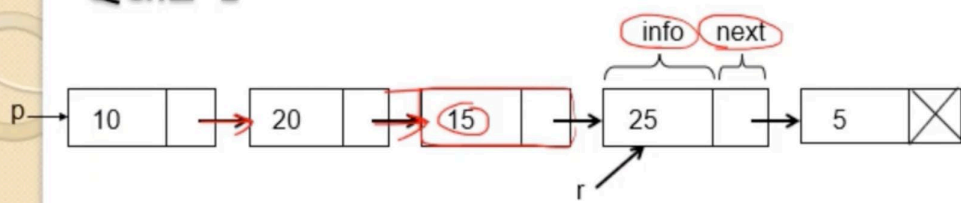
# Array and Linked List (General Linear Lists)



Chapter 02

# วิธีการทำแบบฝึกหัด

**Quiz 1** มีเฉพาะ Data Node



• จงประมวลผลคำสั่งต่อไปนี้ พร้อมวาดภาพผลลัพธ์ของแต่ละคำสั่ง

ข้อ 1.1 ◦ p.next.next.info = 50

ข้อ 1.2 ◦ r.next.next = p.next

◦ p.next.next.next = NULL

◦ p.next = r

◦ p.next.next.next.info = 45

Pointer

DATA STRUCTURES AND ALGORITHMS

42

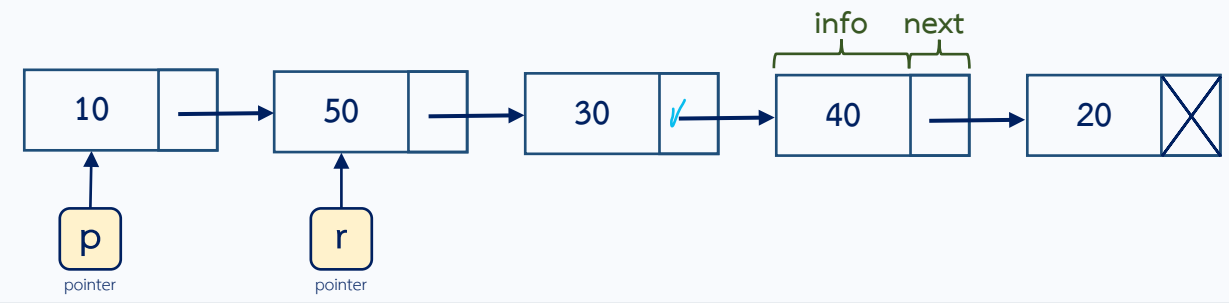
- ให้ประมวลคำสั่งไปเรื่อยๆ ทีละข้อโดยที่ผลลัพธ์ของข้อก่อนหน้าจะเป็นโจทย์ของข้อถัดไป
  - ผลลัพธ์จากข้อ 1.1 จะเป็นตัวโจทย์ของข้อ 1.2
- ต้องเขียนลูกศรชี้ไปแต่ละโหนดข้อมูลและต้องระบุ Pointer ทุกตัวภายในโจทย์

ลิงก์ไฟล์ .pptx

<https://kmitlthailand.sharepoint.com/p/s/22566-ITDataStructuresandAlgorithms/EOzsQjcg11NPm-x6Wij5lj8BK4-gH0Z336tJaQpK7iXpDA?e=Jm61U>

# Exercise 1

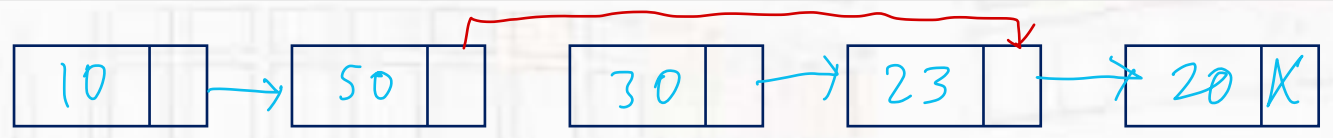
จงประมวลผลคำสั่งต่อไปนี้  
พร้อมวาดภาพผลลัพธ์ของแต่ละคำสั่ง



➤ `r.next.next.info = 23`



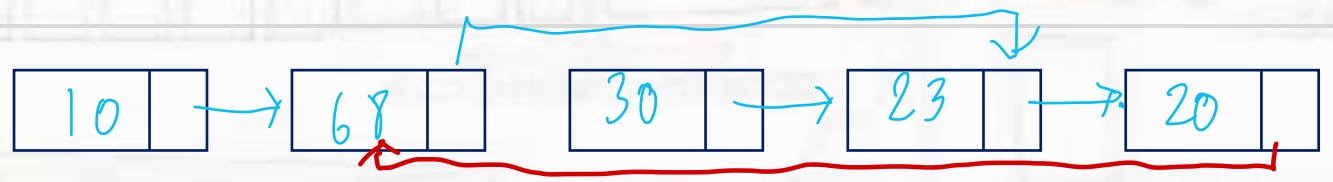
➤ `p.next.next = r.next.next`



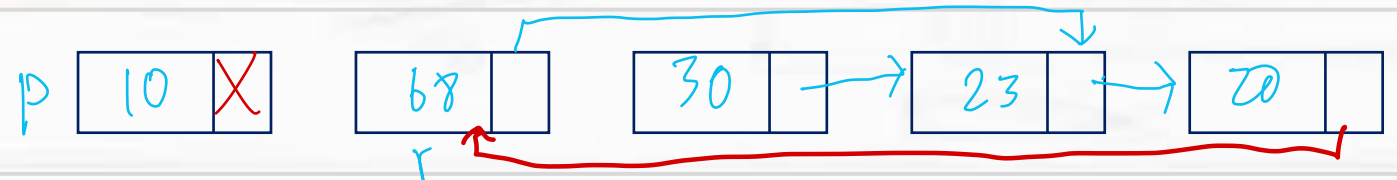
➤ `p.next.info = 68`



➤ `r.next.next.next = p.next`



➤ `p.next = null`





## Exercise 2

จากผลลัพธ์ในข้อก่อนหน้านี้ จงหาผลลัพธ์ของ

```
loop (r.next != null)
  print r.info
  r = r.next
end loop
```

68

23

20

68

23

20

68

23

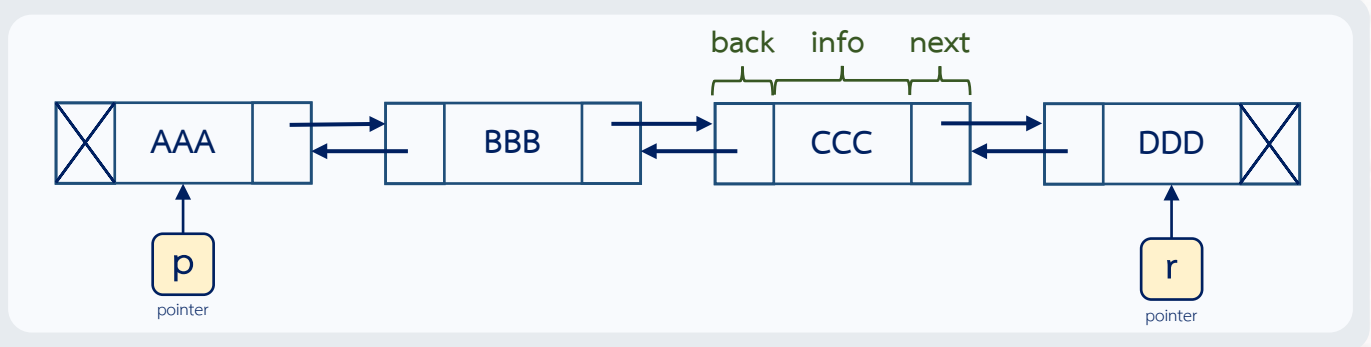
20

⋮

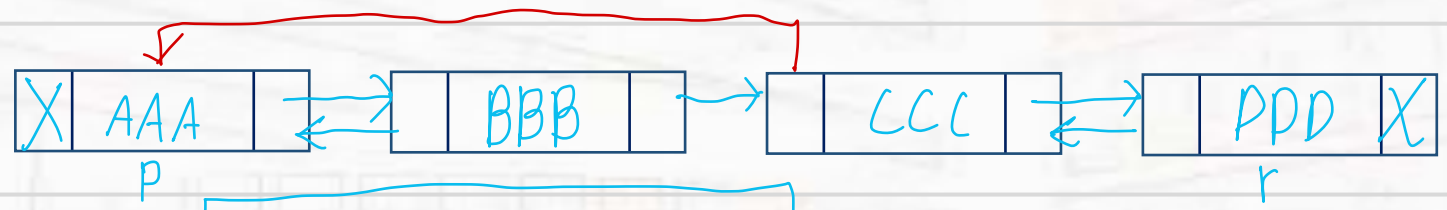
infinite loop \*

# Exercise 3

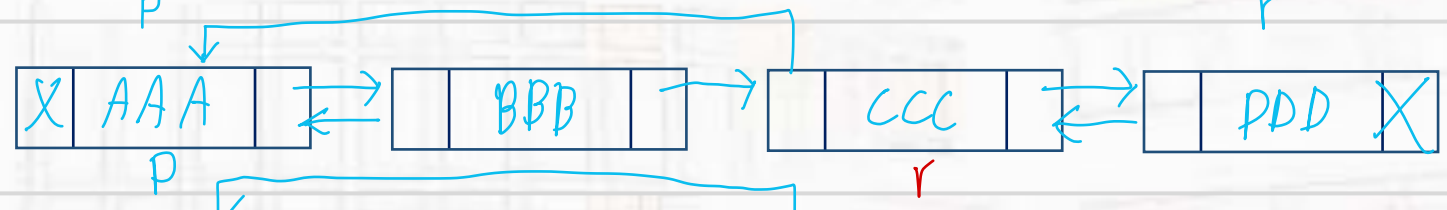
จงประมวลผลคำสั่งต่อไปนี้  
พร้อมวาดภาพผลลัพธ์ของแต่ละคำสั่ง



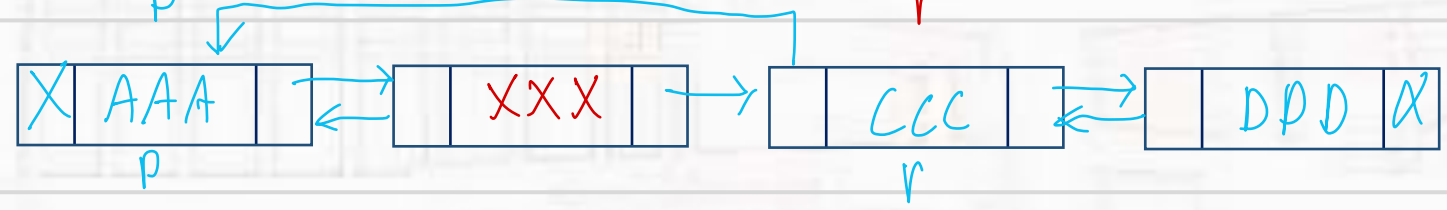
➤ `r.back.back = p`



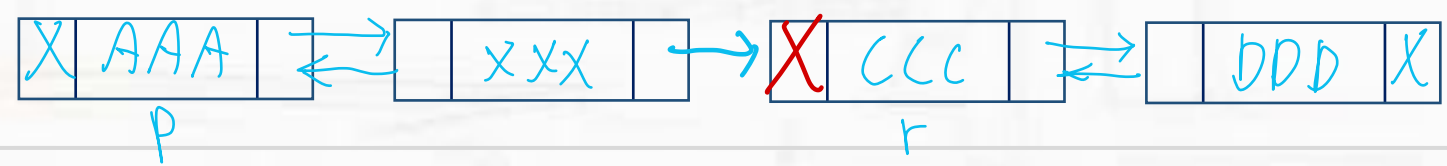
➤ `r = r.back`



➤ `r.back.next.info = 'XXX'`



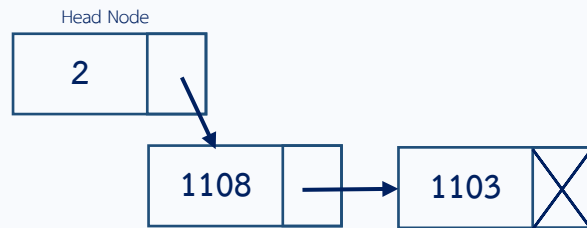
➤ `r.back = p.back`



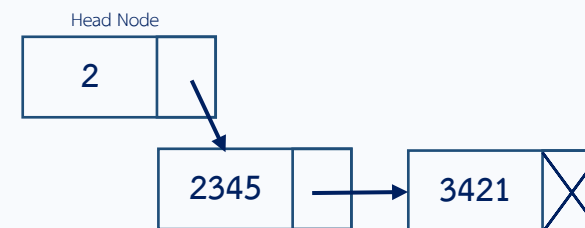
# Exercise 4

จงเขียนอัลกอริทึม append ที่อยู่ในรูปแบบ pseudocode ที่ใช้สำหรับต่อลิสต์ 2 ลิสต์เข้าด้วยกัน

- เช่น `append(list1, list2)` -> list1 และ list2 มีขนาดและข้อมูลเป็นเท่าไรก็ได้

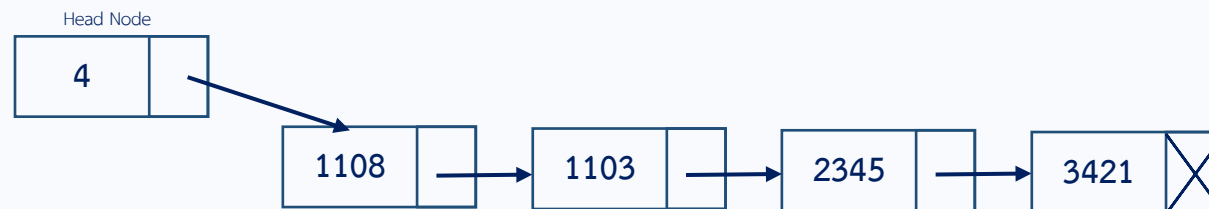


list1



list2

ตัวอย่างผลลัพธ์



list1

```
append (list1, list2)
concat = list1.head
loop (concat.next != null)
    concat = concat.next
end loop
concat.next = list2.head
list1.count += list2.count
end append
```