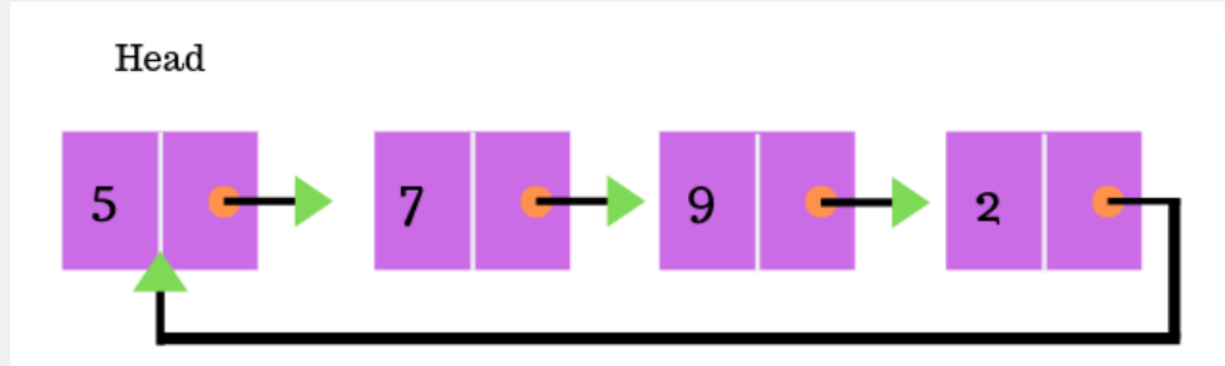


### 3.3 Circularly linked list and circular doubly linked lists

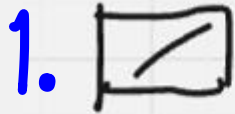
**Circular linked list** คือ linked list ชนิดหนึ่งที่แตกต่างกันจากการโยงปกติ ตรงที่โหนดสุดท้ายของรายการไม่ได้ชี้ไปยังค่า NULL แต่จะชี้กลับไปยังโหนดแรกของ list ทำให้เกิดลักษณะเป็นวงกลม



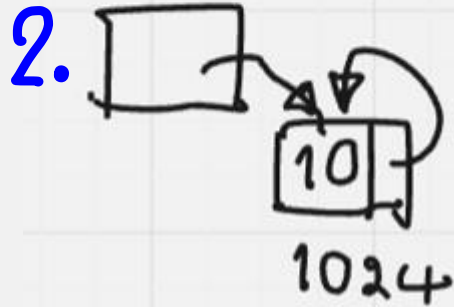
Head

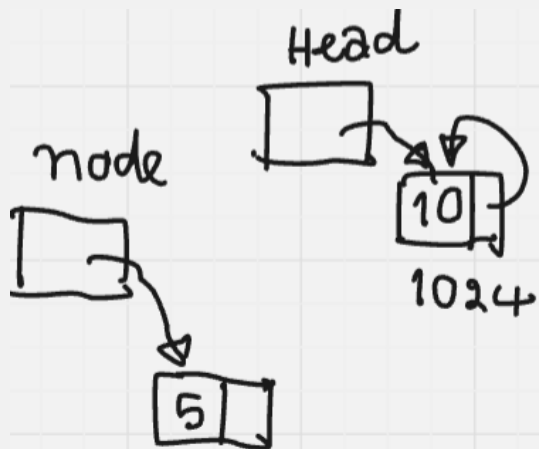
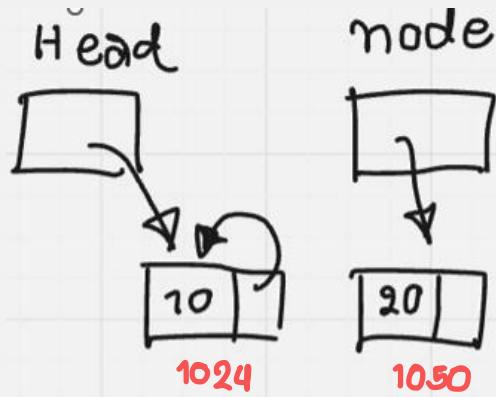
if (head == NULL)

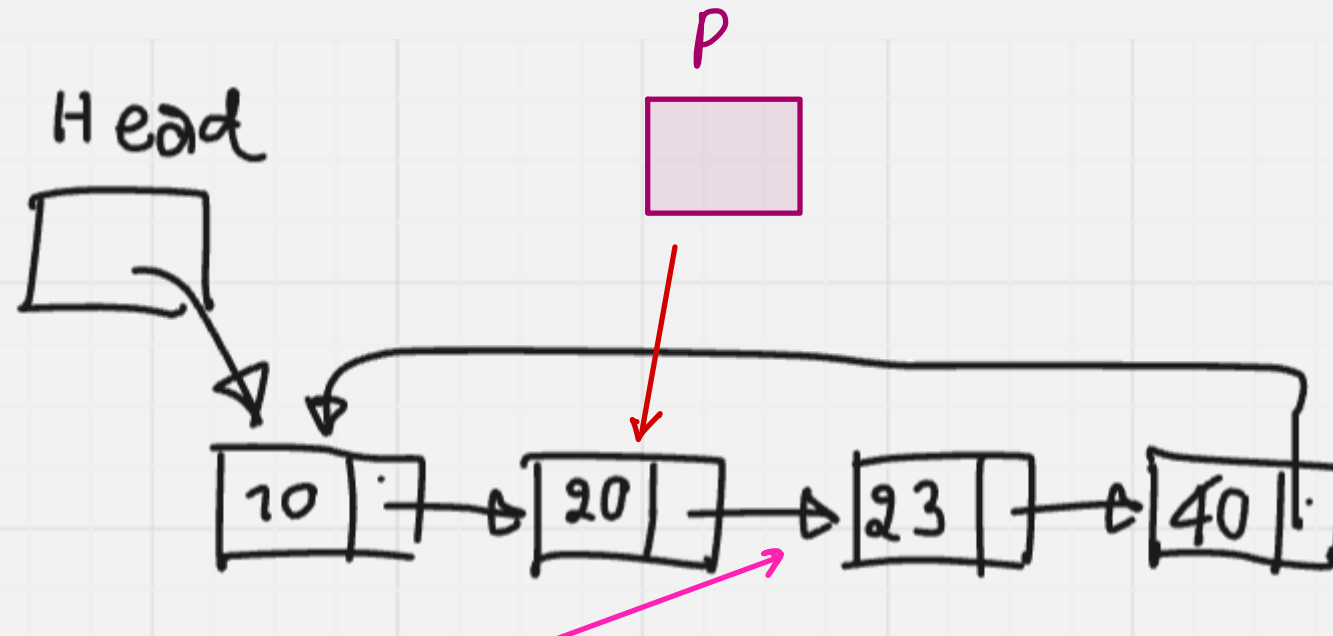
head = new struct record;



Head



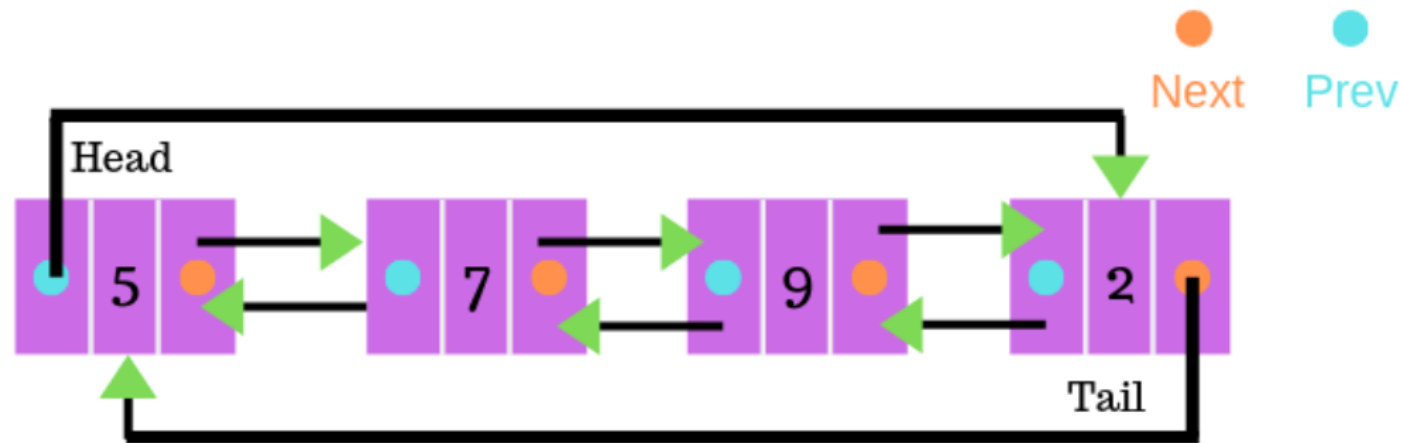


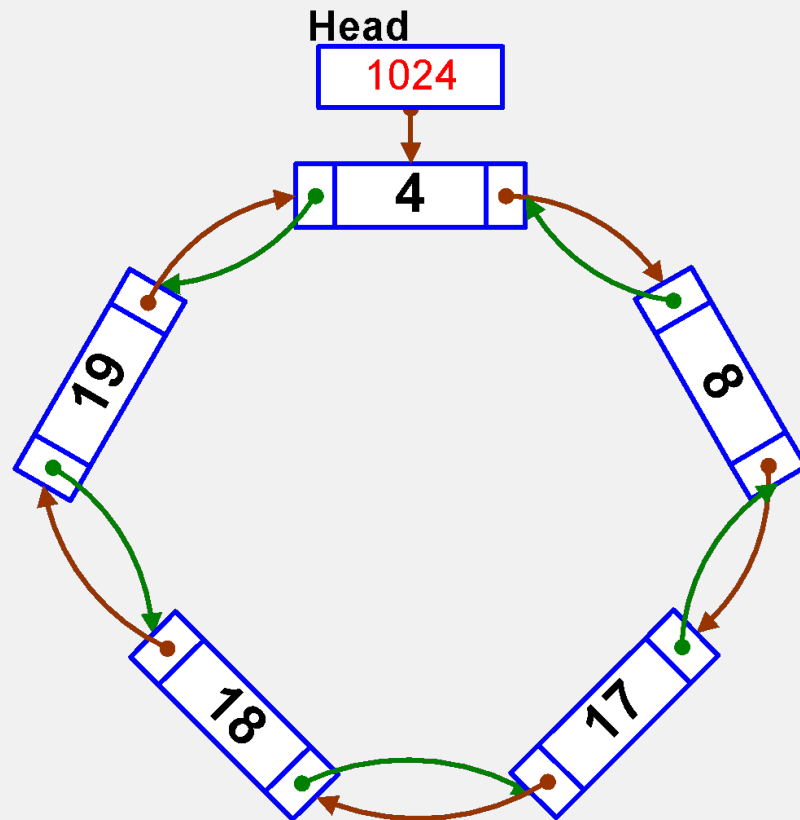


node  
22

$\text{node} \rightarrow \text{next} = \text{p} \rightarrow \text{next};$   
 $\text{p} \rightarrow \text{next} = \text{node};$

**Circularly doubly linked lists** : A popular convention is to have the last cell keep pointer back to the first. This can be done with or without header (If the header present, the last cell point to it.)







## การบ้าน

1. เก็บเลขจำนวนเต็มขนาดไม่จำกัด ให้นิสรับข้อมูลเป็นเลขจำนวนเต็มขนาดไม่จำกัด
- 2 ค่าใส่ linked list โดยมี p1 ชี้ตัวเลขตัวที่ 1 และ p2 ชี้ตัวเลขตัวที่ 2 (ตัวเลขนี้ใหญ่มากจนไม่สามารถเก็บลงในตัวแปร int, float, long double ได้ จึงต้องนำแต่ละ digit มาเก็บลง linked list

=====

MENU

=====

1. Input p1
  2. Input p2
  3. Add
  4. Exit
- Please choose >

ถ้าเลือกข้อ 1

Input : 1234

P1 = 1234

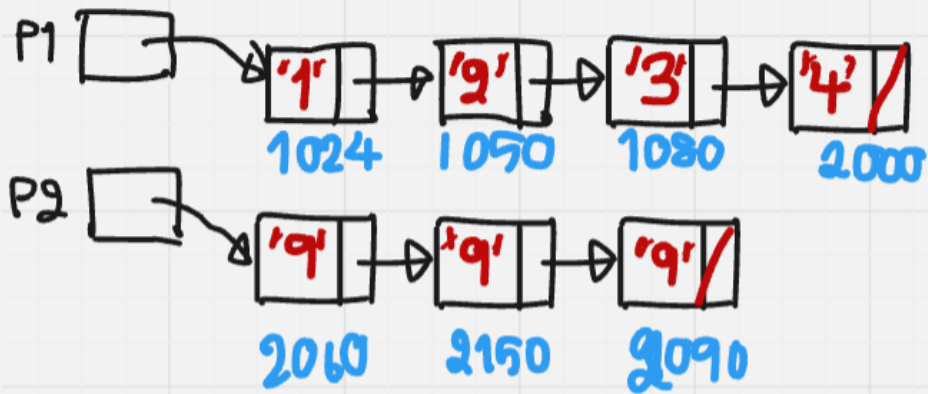
ถ้าเลือกข้อ 2

Input : 899

P2 = 899

ถ้าเลือกข้อ 3

Output = 2133



$$\begin{matrix} x & y \\ \boxed{4} & \boxed{9} = 13 \end{matrix}$$

$$z = x + y;$$

$$\text{cout} \ll z / 10;$$

$$\begin{matrix} \text{tod} \\ \boxed{1} \end{matrix} \quad \text{tod} = z / 10$$





## ตัวอย่าง

```
#include <iostream>
using namespace std;
int main()
{
    char x=5, y=1, unit;
    int a=x;
    int b=y;
    cout << "a=" << a << " b=" << b << endl;
    cout << "a+b=" << a+b;
}
```

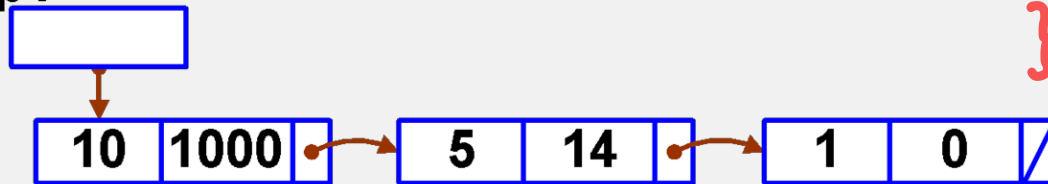
## 3.4 Examples

### 3.4.1 The polynomial ADT

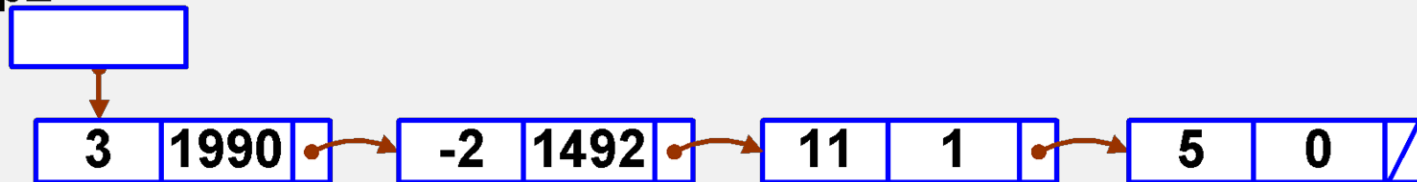
$$p1(x) = 10x^{1000} + 5x^{14} + 1$$

$$p2(x) = 3x^{1990} - 2x^{1492} + 11x + 5$$

p1



p2

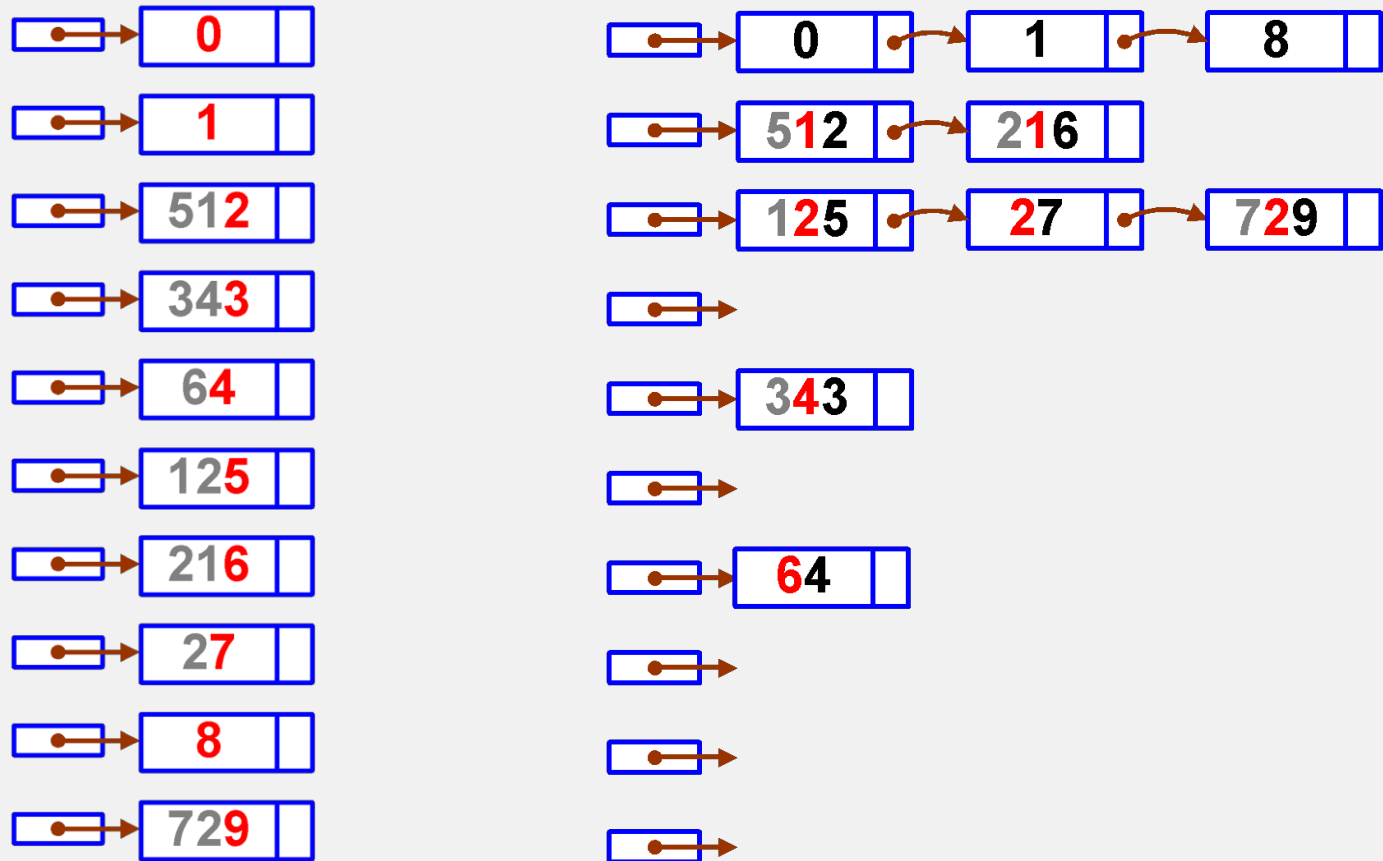


```

struct record{
    int con;
    int pow;
    struct record *next;
};
  
```

## 3.4.2 Radix Sort

Input 64, 8, 216, 512, 27, 729, 0, 1, 343, 125



### 3.4.3 Multilists *array 2 ଭଳି*

