

polynomial expression : 다항식.

리스트 응용문제: 다항식 덧셈

- $p(x) = 4x^4 + 3x^2 + 3$, $q(x) = 3x^3 + 4x^2 + 2x + 1$ 를 더하면 $p(x) + q(x) = 4x^4 + 3x^3 + 7x^2 + 2x + 4$ 가 된다.
- 이 연산을 수행하는 프로그램을 만들어 보자.

- 다항식 $p(x)$ 를 저장하는 자료구조는 $[(4,4), (3,2), (3,0)]$
- add method: 리스트 자료구조에 $(coef, x)$ 형식의 서브 리스트를 추가한다.
- peek method: 0번째 리스트 원소를 리턴한다.
- pop method: 0번째 리스트 원소를 제거 리턴한다.
- size method: 리스트의 길이를 리턴한다.
- print method: 다항식 형태를 프린트 한다.

다항식을 리얼하는 자료구조는 list 이고
해당 리스트 내 원소들의 class type도
list이다.

```
class Poly:
    def __init__(self):
        self.eq = []
    def add(self, coef, k):
        self.eq.append((coef, k))
    def peek(self):
        return self.eq[0]
    def pop(self):
        return self.eq.pop(0)
    def size(self):
        return len(self.eq)
    def print(self):
        for term in self.eq:
            print(term[0], 'x', term[1], '+', end=' ')
        print()

p = Poly()
p.add(4, 4)
p.add(3, 2)
p.add(3, 0)

q = Poly()
q.add(3, 3)
q.add(4, 2)
q.add(2, 1)
q.add(1, 0)

p.print()
q.print()

# print(a, peek())
# print(a, peek())

# print(a, pop())
# print(a, pop())

# p.print()
# q.print()
```

* 지수가 큰 항부터
차례대로 add된다.

→ 가로 방향으로 print 되어진 것들이
표시되어짐.

4 x^4 + 3 x^2 + 3 x^0 +
3 x^3 + 4 x^2 + 2 x^1 + 1 x^0 +

다항식 덧셈 알고리즘

- Step 0: 덧셈 결과를 기록할 빈 다항식을 만든다.
- Step 1: 두개의 다항식에서 최고 차수를 찾고 최고 차수가 있는 리스트에서 팝하여 결과 다항식에 추가한다. 만약, 최고 차수를 가지는 항이 두개 다항식에 공통으로 존재하면 계수를 합하여 add한다.
- Step 2: 한쪽 다항식이 empty 할 때까지 Step 1을 수행한다.
- Step 3: 비어 있지 않은 다항식의 항을 결과 다항식에 add 한다.

해당 parameter에는 '다항식' 객체가 인자값으로 들어간다.
두 다항식을 더한 결과의 객체도 class type이 당연히 '다항식'이어야 한다.

핵심!!
pop한 결과를
'다항식'에
추가함.

'a' 다항식이 남아있는 최고차항이 'b' 다항식에 남아있는
최고차항보다 클 때 => 'a'의 최고차항이 'c' 다항식에
그대로 추가됨.

'b' 다항식의 모든 항들이 pop되고,
'a' 다항식의 요소들만 남아 있을 때.
'a' 다항식의 모든 항들이 pop되고,
'b' 다항식의 요소들이 남아 있을 때.

```
c = polyAdd(p, q)
c.print()

4 x^4 + 3 x^3 + 7 x^2 + 2 x^1 + 4 x^0 +
```

* 더하는 두 다항식의 최고차항 비교는 더하는 라인이 끝날때까지
진행하기 때문에, 'while' 문을 사용해야함.

① 두 다항식 전부 항들을 가지고 있을 때
- 최고차항이 다를 때

↳ 최단화한 때.

② 두 다항식 중 하나의 다항식만 한도를 가지고 있을 때