



[23.01.14] XY Polynomial

TSxyPolynomial

김 태 현

문 제

X와 Y의 변수로 이루어진 다항식이 있다.

다항식은 0~9의 숫자와 X, Y 변수, +, -, ^ 연산자로 이루어져 있다.

- 1) 곱하기 연산자는 생략되고 수와 수를 곱하는 경우는 없다.
수와 변수들이 곱하는 경우 수가 가장 먼저 오고 그 다음 변수들이 온다.
ex) $2 * X \Rightarrow 2X$, $X * 2 * Y \Rightarrow 2XY$ or $2YX$, $2 * 7$
- 2) 지수 연산자는 ^로 표현되고 지수는 오직 숫자로만 구성되어 있다.
ex) $Y^2 = Y^2$, X^{-1} , Y^X
- 3) 다항식이 새롭게 주어질 때, 각 항의 계수와 지수는 한자리 수이다.
또한 계수가 0으로 주어지는 경우는 없다.
하지만, 다항식과 다항식을 연산한 후에 계수와 지수는 두 자리 이상 될 수 있다.
- 4) 다항식끼리의 연산은 더하기 빼기 곱하기 세 가지가 존재한다.
- 5) 주어진 다항식의 맨 앞에 +, - 부호부터 시작하는 경우가 없다.
- 6) 각 항의 변수들의 순서는 제약이 없지만 각 항에서 X, Y가 두 번 이상 나오는 경우는 없다.
ex) $X^2Y = YX^2$, $5XX$, $2YX^2Y$
- 7) 다항식에서 같은 종류의 항들은 같이 묶어서 간단히 한다.
ex) $XY + 2XY - 3 \Rightarrow 3XY - 3$

[제약사항]

1. assign() $\leq 1,000$
2. compute() $\leq 9,000$ (op=2인 경우 1,000회 이하)
3. 모든 함수 $\leq 50,000$
4. 모든 다항식의 항의 개수 총합 $\leq 200,000$

void init()

초기화 함수로 TC처음 1회 호출된다.

void assign(char name[], char mPoly[])

이름이 name인 mPoly 다항식이 주어진다.

이름은 10자리 이하 소문자, 다항식은 100자리 이하로 주어진다.

void compute(char nameR[], char nameA[], char nameB[], int op)

nameA, nameB 다항식을 연산하여 nameR 이름의 다항식을 생성한다.

op가 0,1,2인 경우 각각 더하기, 빼기, 곱하기 연산을 진행한다.

계산 결과 다항식에서 각 항의 X, Y 지수가 모두 1,000 이하이고 항의 개수가 500개 이하임이 보장된다.

int getCoefficient(char name[], int degreeX, int degreeY)

name 다항식에서 X, Y 지수가 degreeX, degreeY인 항의 계수 값을 찾는다.

찾은 계수의 값을 10,000으로 나눈 나머지를 반환한다.

만약 나머지가 음수인 경우 그 나머지 값에 10,000을 더한 값을 반환한다.

또한 해당하는 항이 없다면 0을 반환한다.

int calcPolynomial(char name[], int mX, int mY)

name 다항식의 X, Y 값을 mX, mY로 대입한 결과를 반환한다.

결과는 위 함수와 동일한 형태로 반환한다.

다항식 관리

- 이름에 따른 다항식 접근 필요
- 특정 항의 계수 접근 필요
- 같은 종류의 항들끼리 계산 필요

※ 가장 간편한 방법

`unordered_map<이름, 다항식>`

이름: string

다항식: `unordered_map<{X지수, Y지수}, 계수>`

* $\{X지수, Y지수\} \Rightarrow (X지수 * 1001 + Y지수)$ 로 표현 시 간편

`=> unordered_map<string, unordered_map<int, int>>`

다항식 관리

계산 결과를 반환하는 `getCoefficient()`, `calcPolynomial()` 에서
0 ~ 9,999 범위만 취급하므로 다항식을 저장할 때도 해당 범위로 표현한다.

나머지 연산(%) 분배 법칙

1. $(A * B) \% M = ((A \% M) * (B \% M)) \% M$
2. $(A + B) \% M = ((A \% M) + (B \% M)) \% M$
3. $(A - B) \% M = ((A \% M) - (B \% M) + M) \% M$

* 나눗셈은 적용 안됨

이 문제의 경우

- 1) $M = 10,000$ 으로 설정
- 2) A, B 는 항상 0 ~ 9,999 이므로 모든 연산자 op 에 대해 아래와 같이 일괄적으로 계산 가능
 $\Rightarrow (A \text{ op } B + M) \% M$

assign() 다항식 식별

다항식 구성 요소&처리

1) +, -

- 이전 항의 완성, 새로운 항의 시작
- +면 계수 1, -면 계수 -1로 설정
- 뒤에 숫자가 나온다면 "계수*=숫자"

2) ^

뒤에 반드시 숫자가 나옴

3) X, Y

뒤에 ^가 나온다면 ^뒤의 숫자가 해당 변수의 지수를 표현
뒤에 ^가 나오지 않는다면 해당 변수의 지수는 1

4) 1~9

- 1) 계수 or X의 지수 or Y의 지수 중 한가지를 표현

구현

- c : 계수 x : X의 지수 y : Y의 지수
type : 뒤에 나올 숫자가 어떤 값을 의미하는지 표현
0=계수, 1=X의 지수, 2=Y의 지수
- 계수는 0~9,999로 저장
- 다항식 탐색
 1. +, -
 - 이전 항 계수 업데이트
 - type=x=y=0
 - c=1 or -1
 2. X : type = 1, x=1
 3. Y : type = 2, y=1
 4. num(1~9) : type에 따라 c,x,y 에 *=num
 5. ^ : 무시
- + 마지막 항 계수 업데이트

compute() 다항식 계산

새로운 다항식(`unordered_map`)을 생성하여 계수를 업데이트 해나간다.

1) +

- 1) A의 모든 항을 C에 추가
- 2) B의 모든 항을 C에 추가(동일 항이 있다면 계수 더하기)

2) -

- 1) A의 모든 항을 C에 추가
- 2) B의 모든 항의 계수를 -로 하여 C에 추가(동일 항이 있다면 계수 빼기)

3) *

(A의 항, B의 항) 모든 쌍에 대해 지수는 더하고 계수는 곱하여 C에 추가
계산 결과 같은 항들이 나온다면 계수 더하기

※ 모든 계수는 역시 0~9,999 범위로 나오게 한다.

calcPolynomial()

calcPolynomial() $\leq 50,000$

다항식 항의 개수 ≤ 500

$0 \leq X, Y$ 지수 $\leq 1,000$

$-4 \leq mX, mY \leq 4$

- 매번 각 항의 지수들까지 계산하는건 비효율적이다.
- 어차피 주어지는 mX, mY 의 종류가 9개, 지수는 1001개 뿐이므로 Lookup Table을 기록하여 활용한다.
$$\text{LUT}[\text{val}+4][\text{exp}] = \text{val}^{\text{exp}} \quad (0 \sim 9,999)$$
- 해당 다항식의 모든 항을 linear search 하여 바로 계산해줄 수 있다.

감사합니다

