台北市立
松山高級中學

114 學年度
學科能力競賽
程式設計集訓講義

Chen Haning 2025.07

目次

數值計算篇

科學家和工程師所做的計算，大都與數值有關，要解的是數值問題（numerical problem）。以電子計算機解數值問題，就是將實數資料輸入，利用電子計算機做快速的加減乘除等運算，將求出的實數結果輸出，這樣的演算過程，統稱爲數值計算（numerical computation）。

數值資料的準確度，受到表示它的位元數所左右，因此一般數值計算多少都有誤差。影響誤差大小的因素很多，例如計算前所收集之數值資料的準確度，計算機內部表示法所用的位元數；每次運算產生的誤差（如四捨五入）；多次運算所累積而衍生的更大誤差；以有限的執行步驟不能夠無限制的提高精確度等，可見數值計算的精確度是很難預估的。

數值計算不但求快，而不產生太大的誤差也是重要的課題。本章並不深入探討數值計算的誤差，主要是介紹一些基本的數值問題，以及解決它們的數值計算方法。

9-1 求多項式的函數值

多項式是最基本的函數，n 次多項式可以寫成：

$$p(x)=a_0x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n$$

求多項式的函數值就是把係數 $a_0,a_1,\cdots$，a_n 和變數 x 的值輸入，求出p(x)的值。最平常的方法是求出每一項的值，並累加起來。如果以陣列A儲存各項的係數值，以X儲存變數值，以Y代表X的乘冪，而以P代表結果，則其演算法則如下：

輸入多項式的次數N
輸入係數值到 A(0), A(1),…, A(N)
輸入變數值到X

求多項式函數值

設定P為A(N)

設定Y為X

使用 FOR 迴路（I由N－1遞減到0）重覆底下步驟:

  P＝P＋A(I)＊Y

  Y＝Y＊X

輸出P

在這個計算方法當中，每求一項就要做兩次乘法，總共要做 2n 次乘法，其實另外有一個更快的方法，總共只要做n次乘法就可以了。如果把多項式寫成

$$p(x)=((\cdots((a_0x+a_1)x+a_2)x+\cdots)x+a_{n-1})x+a_n$$

利用以上的式子，由 a_0 開始，每次乘上x後加上下一項的係數，直到乘上x再加上a_n，計算就完成了，其演算法則如下:

輸入多項式的次數N

輸入係數值到 A(0), A(1),…, A(N)

輸入變數值到X

設定P為 A(0)

使用FOR迴路（I由1遞增到N）重覆底下步驟:

  P＝P＊X＋A(I)

輸出P

程式如下:

```
100  DIM A(10)
200  INPUT "DEG OF POLYNOMIAL";N
210  FOR I=0 TO N
220       INPUT "COEFFICIENT";A(I)
230  NEXT I
240  INPUT "VALUE OF X=";X
300  P=A(0)
310  FOR I=1 TO N
320       P=P*X+A(I)
330  NEXT I
400  PRINT "VALUE OF POLYN AT X =";X;" IS  " ;
410  PRINT P
500  END
```

如果多項式爲 $P(x)=x^3+3x^2+3x+1$，要求 $P(1.5)$的值。則程式的執行情形如下:

```
RUN
DEG OF POLYNOMIAL? 3
COEFFICIENT? 1.0
COEFFICIENT? 3.0
COEFFICIENT? 3.0
COEFFICIENT? 1.0
VALUE OF X=? 1.5
VALUE OF POLYN AT X = 1.5 IS 15.625
OK
```

9-2 求方程式的根

9-2-1 求根的數值問題

一個變數的方程式可以寫成 $f(x)=0$，其中 f 是特定的數學函數，求方程式的根，就是計算出使得 $f(x)$ 等於 0 的所有 x 的值。如果 f 是簡單的函數，可能有公式可以求根，例如 f 爲二次多項式 ax^2+bx+c，則求方程式 $f(x)=0$ 根的公式爲

$x=\dfrac{-b\pm\sqrt{b^2-4ac}}{2a}$，式中除了做加減乘除等運算外，還須要求平方根。當然在 BASIC 裏有內建函數 SQR 可用，但那也是別人事先設計好的程式，求平方根 $\sqrt{d}$ 相當於解方程式 $x^2-d=0$。

如果 f 爲三次或四次多項式，求 $f(x)=0$ 的根雖然仍有公式，但相當複雜，如果照著去設計程式，所得的答案可能誤差很大而不適用。再看 f 爲更高次多項式的情形，這時連一般的公式都沒有了，由此可見實際計算方程式的根不能太依賴數學公式，而必須把它看成數值問題，以計算機數值計算的方法求數值解。

求根的問題可分成兩個部分，首先在定義域裏選定一個區間，保證在那個區間裏有根存在，然後在該區間把根的位置找出來，由於是數值計

算，我們不能夠得到根的確切位置，而只能求到一個近似位置，卽根的近似值。

要選定適當的區間，則必須對函數本身有相當的了解才行。可以在一個足夠大的區間裏求一些函數值，若發現有 x_1, x_2，其函數值 $f(x_1), f(x_2)$ 不同號時（一正一負），我們便知道在 (x_1, x_2) 這個區間裏至少有一個根。這個概念以函數圖來看更清楚，圖 9-1(a) 表示區間裏有一個根，圖 9-1(b) 表示區間裏有一個以上的根。如果區間裏有一個以上的根，就必須再加以分段，使得每一個小區段裏只有一個根。分段的方法是求出區間中函數圖的峯點和谷點，（可以用函數圖加以簡單估計），如圖 9-1(b) 中的 x_3, x_4, x_5, x_6 等點，以它們來晝分出 (x_3, x_4)，(x_4, x_5)，(x_5, x_6) 等三個區間，然後分別找三個根的近似位置。底下介紹兩種在適當區間裏求根的方法。

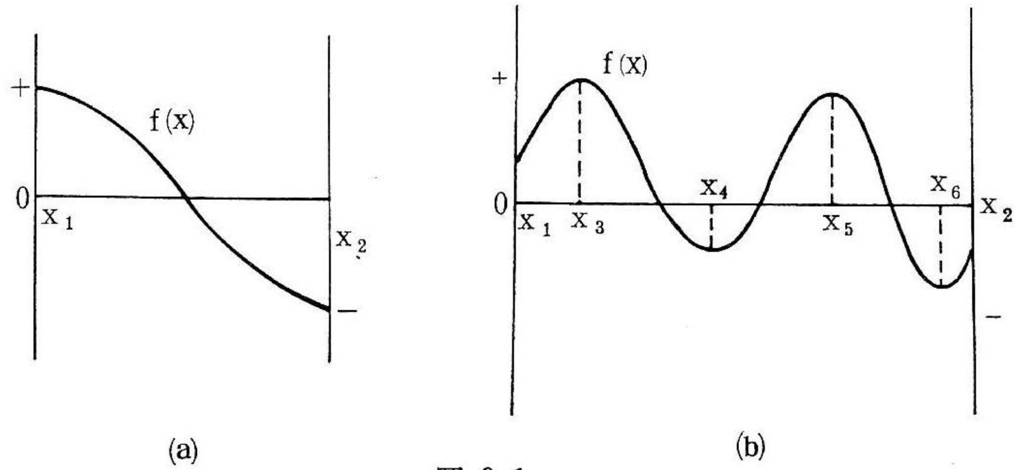

圖 9-1

9-2-2 等分法

等分法就是把區間二等分，決定那一半含有根後，將該半再二等分，如此繼續下去，直到區間的長度相當小爲止，因爲區間越小就表示越接近根的眞正位置。決定往那邊細分則是靠函數值的正負關係，假設 x_3 爲 (x_1, x_2) 的中間點，若 $f(x_1)$ 和 $f(x_3)$ 異號就取區間 (x_1, x_3)；若 $f(x_2)$ 和 $f(x_3)$ 異號就取區間 (x_3, x_2)。圖 9-2 顯示等分法進行的情形：由於 $f(x_2)$ 與 $f(x_3)$ 異號，我們取 (x_3, x_2)，(x_3, x_2) 的中間點爲 x_4，因

$f(x_3)$ 與 $f(x_4)$ 異號，所以取 (x_3, x_4)，依同樣原則再取 (x_3, x_5)，如此繼續下去，漸漸靠近根的確切位置。

演算法則列在下面，其中 F 表示函數 f，寫程式時可自定，P1, P2表示區間左右兩個端點， P3 表示區間的中點，函數值是否同號以其乘積的正負來決定。注意：演算法則中的 P1 永遠小於 P2。

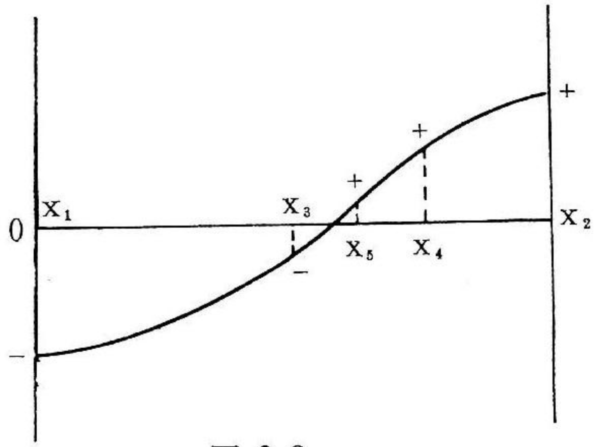

圖 9-2

定義函數 F
輸入 P1, P2
若 F(P1)*F(P2)≦0 則做
　當 P2－P1 不夠小時重覆底下步驟:
　　P3＝(P1＋P2)/2
　若 F(P1)*F(P3)＜0 則做
　　P2＝P3
　否則做
　　P1＝P3
輸出 (P1＋P2)/2
否則輸出“此區間無根”

若將 $f(x)=x^2-2$ 代入上面的演算法則， 則得到一求 $\sqrt{2}$ 近似值的程式。為了避免重覆計算函數值，可以用 V1, V2, V3 分別存 F(P1), F(P2), F(P3) 的值。$\sqrt{2}$ 顯然介於 1 與 2 之間， P1 和 P2 可分別輸入 1 和 2 作為區間端點的初值。

```
100 DEF FNPLOY(X)=X*X-2
110 INPUT"SELECT TWO PROPER END POINTS ";P1.P2
120 V1=FNPLOY(P1)
130 V2=FNPLOY(P2)
140 :
150 IF V1*V2>0 THEN GOTO 230
160     WHILE P2-P1>=.000001
170         P3=(P1+P2)/2
180         V3=FNPLOY(P3)
190         IF V1*V3>0 THEN P1=P3:V1=FNPLOY(P1)
                       ELSE P2=P3:V2=FNPLOY(P2)
200     WEND
210 PRINT"ROOT IS APPROXIMATELY ";(P1+P2)/2
220 GOTO 240
230 PRINT "NO ROOT IN INTERVAL."
240 END
RUN
SELSCT TWO PROPER END POINTS.? 1,2
ROOT IS APPROXIMATELY  1.414214
```

9-2-3 內插法

內插法不以定義域X軸上的 x_1,x_2 爲出發點，而是考慮函數圖上的兩個點 $(x_1,f(x_1))$ 和 $(x_2,f(x_2))$，若把兩點連成一直線，該直線一定和X軸相交於 x_1,x_2 間的一點 x_3，然後可重新選定區間 (x_1,x_2) 或 (x_3,x_2)，繼續做同樣的動作，如此以相交點而不是中間點去逼近根的位置。圖 9-3 表達用內插法求根近似值的過程：由 $(x_1,f(x_1))$ 和 $(x_2,f(x_2))$ 得到交點 x_3，由 $(x_1,f(x_1))$ 和 $(x_3,f(x_3))$ 得到交點 x_4，以此逼近根的位置。

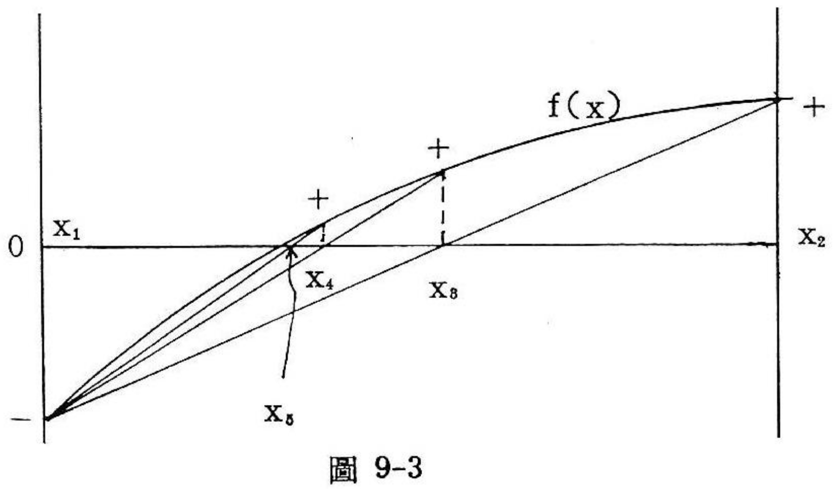

圖 9-3

利用 (P1,F(P1))，(P2, F(P2))， (P3,F(P3)) 三點共線的性質，可求出 P3 的值，其公式爲:

P3=P1−(P2−P1)*F(P1)/(F(P2)−F(P1))

雖然內插法中逼近根的過程和等分法很類似，但 P2−P1 卻不一定會趨近於 0，因此不能以 P2−P1 的大小，而是以 P3 變動的大小來控制迴路的繼續執行與否，整個演算法則描述如下:

```
輸入P1,P2
若 F(P1)*F(P2)≤0 則做
  P3=P1−(P2−P1)*F(P1)/(F(P2)−F(P1))
若 F(P1)*F(P3)<0 則做
  P2=P3
否則做
  P1=P3
重覆底下步驟直到 |P4−P3| 相當小:
  P4=P3
  P3=P1−(P2−P1)*F(P1)/(F(P2)−F(P1))
  若 F(P1)*F(P3)<0 則做
    P2=P3
  否則做
    P1=P3
輸出P3
否則輸出"此區間無根"
```

※9-4 解聯立方程式

現在來考慮如何解下面的聯立方程式:

$$
\begin{array}{l}
a_{11}x_1+a_{12}x_2+a_{13}x_3+\cdots+a_{1n}x_n=a_{1,\ n+1}\\
a_{21}x_1+a_{22}x_2+a_{23}x_3+\cdots+a_{2n}x_n=a_{2,\ n+1}\\
a_{31}x_1+a_{32}x_2+a_{33}x_3+\cdots+a_{3n}x_n=a_{3,\ n+1}\\
\vdots \qquad \vdots \qquad \vdots \qquad\qquad \vdots \qquad \vdots\\
a_{n1}x_1+a_{n2}x_2+a_{n3}x_3+\cdots+a_{nn}x_n=a_{n,n+1}
\end{array}
$$

其中 a_{ij}, $1\leq i, j\leq n$，爲輸入的係數；x_i, $1\leq i\leq n$，爲輸出的解。

如果把任一個方程式換成該方程式減去一個常數乘另一個方程式，並不會改變聯立方程式的解，例如第 i 個方程式可以改成

$$
\begin{array}{rl}
 & a_{i1}x_1+a_{i2}x_2+a_{i3}x_3+\cdots+a_{in}x_n=a_{i,\ n+1}\\
- & c(a_{j1}x_1+a_{j2}x_2+a_{j3}x_3+\cdots+a_{jn}x_n=a_{j,\ n+1})\\
\hline
 & (a_{i1}-ca_{j1})x_1+(a_{i2}-ca_{j2})x_2+(a_{i3}-ca_{j3})x_3+\cdots+\\
 & (a_{in}-ca_{jn})x_n=a_{i,n+1}-ca_{j,n+1}
\end{array}
$$

利用上述的方法，想辦法把原來的聯立方程式改寫成底下的形式（注意方程式左邊的項數越來越少）：

$$
\begin{array}{r}
a_{11}x_1+a_{12}x_2+a_{13}x_3+\cdots+a_{1n}x_n=a_{1,\ n+1}\\
a_{22}x_2+a_{23}x_3+\cdots+a_{2n}x_n=a_{2,\ n+1}\\
a_{33}x_3+\cdots+a_{3n}x_n=a_{3,\ n+1}\\
\vdots \qquad \vdots\\
a_{nn}x_n=a_{n,\ n+1}
\end{array}
$$

這樣一來，從最後的方程式可先算出 x_n 的值爲 $a_{n,n+1}/a_{nn}$。然後由上一個方程式算出 x_{n-1} 的值爲 $(a_{n-1,\ n+1}-a_{n-1,\ n}x_n)/a_{n-1,n-1}$,……直到由第一個方程式求出 x_1 的值。換言之，若已算出 $x_n,x_{n-1},\cdots,x_{i+1}$ 的值，由第 i 個方程式可求出 x_i的值爲 $(a_{i,\ n+1}-a_{i,\ i+1}\ x_{i+1}-\cdots-a_{in}x_n)/a_{ii}$ 。

爲了方便說明，我們用以下的係數陣列來表示聯立方程式:

$$\begin{array}{cccccc}
a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1n} & a_{1,n+1} \\
a_{21} & a_{22} & a_{23} & \cdots\cdots & a_{2n} & a_{2,n+1} \\
a_{31} & a_{32} & a_{33} & \cdots\cdots & a_{3n} & a_{3,n+1} \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
a_{n1} & a_{n2} & a_{n3} & \cdots\cdots & a_{nn} & a_{n,n+1}
\end{array}$$

因爲第一行中的 $a_{21},a_{31},\cdots,a_{n1}$ 都要變成 0 ，而 $a_{i1}-(a_{i1}/a_{11})a_{11}=0$，如果把第 i 列換成該列減掉 (a_{i1}/a_{11}) 乘以第一列，卽 a_{ij} 設定爲 $a_{ij}-(a_{i1}/$ $a_{ij}-(a_{i1}/a_{11})\ a_{1j}$, $2\leq i\leq N$, $2\leq j\leq N+1$，則陣列就變成下面形式:

$$\begin{array}{cccccc}
a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1n} & a_{1,n+1} \\
0 & a_{22} & a_{23} & \cdots\cdots & a_{2n} & a_{2,n+1} \\
0 & a_{32} & a_{33} & \cdots\cdots & a_{3n} & a_{3,n+1} \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
0 & a_{n2} & a_{n3} & \cdots\cdots & a_{nn} & a_{n,n+1}
\end{array}$$

接著在框起來的部分又可重覆上述的步驟，第三列以後都減去 (a_{i2}/a_{22}) 乘以第二列， 卽 a_{ij} 設定爲 $a_{ij}-(a_{i2}/a_{22})a_{2j}$，$3\leq i\leq N$，$3\leq j\leq N+1$，陣列變成:

$$\begin{array}{cccccc}
a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1n} & a_{1,n+1} \\
0 & a_{22} & a_{23} & \cdots\cdots & a_{2n} & a_{2,n+1} \\
0 & 0 & a_{33} & \cdots\cdots & a_{3n} & a_{3,n+1} \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
0 & 0 & a_{n3} & \cdots\cdots & a_{nn} & a_{n,n+1}
\end{array}$$

如此，每次處理的陣列範圍都縮小一點，重覆 n−1 次後，整個陣列就變成所要的形式:

$$
\begin{matrix}
a_{11} & a_{12} & a_{13} & \cdots\cdots a_{1n} & a_{1,n+1} \\
0 & a_{22} & a_{23} & \cdots\cdots a_{2n} & a_{2,n+1} \\
0 & 0 & a_{33} & \cdots\cdots a_{3n} & a_{3,n+1} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots\cdots a_{nn} & a_{n,n+1}
\end{matrix}
$$

若以二維陣列A(N, N+1) 代表係數，而以一維陣列 X(N) 代表聯立方程式的解，演算法則初步構想如下:

輸入係數到陣列A
用FOR迴路（K由1遞增到N−1）重覆底下步驟:
　更改第K+1列到第N列
X(N)=A(N, N+1)/A(N,N)
用FOR迴路（I由N−1遞減到1）重覆底下步驟:
　計算X(I)
輸出陣列X

「更改第K+1列到第N列」可寫成

用FOR迴路（I由K+1遞增到N）重覆底下步驟:
　更改第I列

「更改第I列」部分又可進一步改寫成

用FOR迴路（J由K+1遞增到N+1）重覆底下步驟:
　A(I,J)=A(I,J)−(A(I,K)/A(K,K))*A(K,J)

而「計算X(I)」則可改寫成

SUM=0
用FOR迴路（J由I+1遞增到N）重覆底下步驟:

```
  SUM=SUM+A(I, J)*X(J)
X(I)=(A(I, N+1)-SUM)/A(I, I)
```

整個演算法則為：

```
輸入係數到陣列A
用FOR迴路（K由1遞增到N-1）重覆底下步驟:
  用FOR迴路（I由K+1遞增到N）重覆底下步驟:
    用FOR迴路（J由K+1遞增到N+1）重覆底下步驟:
      A(I,J)=A(I,J)-(A(I,K)/A(K,K))*A(K,J)
X(N)=A(N, N+1)/A(N,N)
用FOR迴路（I由N-1遞減到1）重覆底下步驟:
  SUM=0
  用FOR迴路（J由 I+1 遞增到N）重覆底下步驟:
    SUM=SUM+A(I, J)*X(J)
  X(I)=(A(I, N+1)-SUM)/A(I, I)
輸出陣列X
```

問　題

1. 以內插法寫一程式，以求$\sqrt{n}$的近似值，n為任意正整數。
2. 寫一程式，以求正弦函數 SIN 的近似值。

   註：$SIN(x)=\sum_{n=0}^{\infty}\frac{(-1)^n x^{2n+1}}{(2n+1)!}=\frac{x}{1!}-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\cdots$
3. 寫一程式，以便能快速求得 x^n，設 $x=2.7$，n 為任意正整數。

   註：若將 n 以二進位表示為 $a_m a_{m-1}\cdots a_0$ （$n=a_m 2^m+a_{m-1}2^{m-1}+\cdots+a_1 2+a_0$），則 $x^n=x^{a_m 2^m}x^{a_{m-1}2^{m-1}}\cdots x^{a_1 2}x^{a_0}$，所以如果能依序求出 $a_0,a_1,a_2,\cdots,a_m$，則可在 m 步裏算出 x^n，這裏的 m 並不需要預先知道。
4. 將解聯立方程式的演算法則改寫成程式。

漸進式 9-3數值計算

寫一程式，以便能快速求得 x^n ，設x=2.7，n 為任意正整數。

註：若將n以二進位表示為 $a_m a_{m-1} \ldots a_0$（$n = a_m 2^m + a_{m-1} 2^{m-1} + \ldots + a_1 2 + a_0$）

則 $x^n = x^{a_m 2^m} x^{a_{m-1} 2^{m-1}} \ldots x^{a_1 2} x^{a_0}$ ，所以如果能依序求出 $a_0, a_1, a_2 \ldots, a_m$, 則可在m步裡算出 x^n ，這裡的m並不需要預先知道。

解:

$$x^n = x^{a_m 2^m} x^{a_{m-1} 2^{m-1}} \ldots x^{a_1 2} x^{a_0}$$

$$= ((((x^{a_m})^2 \cdot x^{a_{m-1}})^2 \cdot x^{a_{m-2}})^2 \cdot x^{a_{m-3}})^2 \ldots x^{a_0}$$

令 $p = x^{a_m}$

重覆m→1次

$p = p^2 \cdot x^{a_{m-1}}$

例：求 x^{11}

解： $x^{11} = x^{1011_2} = x^{1 \cdot 2^3} x^{0 \cdot 2^2} x^{1 \cdot 2^1} x^{1 \cdot 2^0}$

$$= (((x^1)^2 \cdot x^0)^2 \cdot x^1)^2 \cdot x^1$$

驗算：

$$\begin{aligned}
&(((x^1)^2 \cdot x^0)^2 \cdot x^1)^2 \cdot x^1 \\
&= ((x^2 \cdot 1)^2 \cdot x^1)^2 \cdot x^1 \\
&= (x^4 \cdot x^1)^2 \cdot x^1 \\
&= (x^5)^2 \cdot x^1 \\
&= x^{10} \cdot x^1 \\
&= x^{11}
\end{aligned}$$

指數律

$x^{ab} = (x^a)^b$

$x^{a+b} = x^a \cdot x^b$