C basic language

임베디드스쿨 2기
Lv1과정
2021. 04. 23
김효창

- 목 차 -

1. 벡터

Radian 호도법

호의 길이를 이용해서 각도를 표시

1 radian x $180/\pi$ = 57.296°

1° x $\pi/180$ = 0.01745 radian

자동차 주행 중 운전 손잡이를 오른쪽으로 10°돌리다.

(사람은 degree , **컴퓨터는 radian**)

θ	0°	30°	45°	60°	90°	120°	180°	270°	360°
radian	0	π/6	π/4	π/3	π/2	2π/3	π	3π/2	2π
sinθ	0	1/2	$\sqrt{2}/2$	$\sqrt{3}/2$	1	$\sqrt{3}/2$	0	-1	0
cosθ	1	$\sqrt{3}/2$	$\sqrt{2}/2$	1/2	0	-1/2	-1	0	1
tanθ	0	$1/\sqrt{3}$	1	$\sqrt{3}$	∞	$-\sqrt{3}$	0	∞	0
cotθ	∞	$\sqrt{3}$	1	$1/\sqrt{3}$	0	$-1/\sqrt{3}$	∞	0	∞
secθ	1	$2/\sqrt{3}$	$\sqrt{2}$	2	∞	-2	-1	∞	1
cosecθ	∞	2	$\sqrt{2}$	$2/\sqrt{3}$	1	$2/\sqrt{3}$	∞	-1	∞

1. 벡터

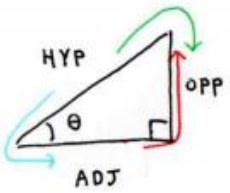
Scalar

값이 단일 실수로 표시될 수 있는 수량
크기는 있지만 방향은 없다
예) Byte , 면적 , 부피, 시간 , 거리 , 속력 (거리/시간) , 질량 , 에너지 , 밀도(농도) , 길이 , 온도 , 기본 대수에서 사용하는 x , y , z

Vector

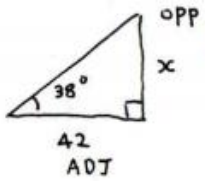
크기와 방향이 모두 있는 것
예) 변위 , 속도(변위/시간) , 무게 , 가속도 , 축전지의 양극에서 음극으로의 힘 , 압력 중력, 자기장, 케이블의 전압 기울기, 납땜 인두 팁의 온도 기울기
벡터 내적 결과는 숫자 (스칼라 값으로 변경) , $\|A\| \cdot \|B\| \cos\theta$
벡터 외적 결과는 i + j + k (벡터 값으로 변경) , $\|A\| \cdot \|B\| \sin\theta$

1. 벡터



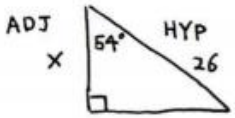
$$\sin\theta = \frac{OPP}{HYP} \qquad \tan\theta = \frac{OPP}{ADJ}$$

$$\cos\theta = \frac{ADJ}{HYP}$$



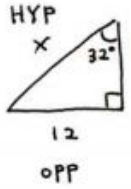
$$42\tan(38^\circ) = \frac{x}{42} \times 42$$

$$X = 42\tan(38^\circ) = 32.8$$



$$\cos(54^\circ) = \frac{X}{26}$$

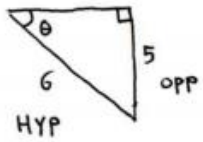
$$X = 26\cos(54^\circ) = 1528$$



$$\frac{\sin(32^\circ)}{1} = \frac{12}{X}$$

$$12 = X\sin\theta$$

$$X = \frac{12}{\sin(32^\circ)} = 22.64$$



$$\sin\theta = \frac{5}{6}$$

$$\theta = \sin^{-1}\left(\frac{5}{6}\right)$$

$$\theta = 56.44^\circ$$

$$\sin(30^\circ) = \frac{1}{2} \qquad \sin(60^\circ) = \frac{\sqrt{3}}{2}$$

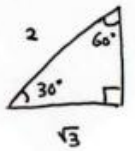
$$\cos(30^\circ) = \frac{\sqrt{3}}{2} \qquad \cos(60^\circ) = \frac{1}{2}$$

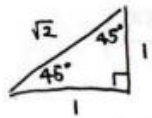
$$\tan(30^\circ) = \frac{1}{\sqrt{3}}\frac{\sqrt{3}}{\sqrt{3}} = \frac{\sqrt{3}}{3} \qquad \tan(60^\circ) = \frac{\sqrt{3}}{1} = \sqrt{3}$$

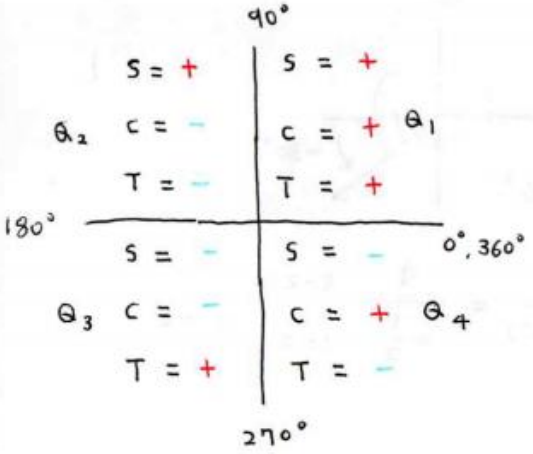
$$\sin(45^\circ) = \frac{1}{\sqrt{2}} = \frac{\sqrt{2}}{2}$$

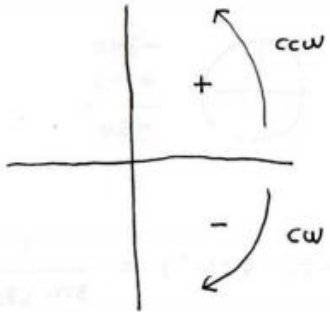
$$\cos(45^\circ) = \frac{\sqrt{2}}{2}$$

$$\tan(45^\circ) = 1$$



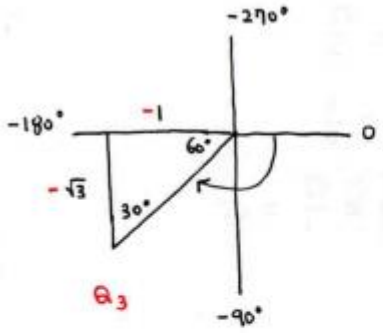






A	S	T	C
Q_1	Q_2	Q_3	Q_4

1. 벡터

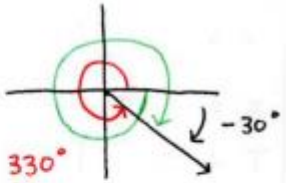


$\tan(-120°)\ \frac{-\sqrt{3}}{-1} = \sqrt{3}$

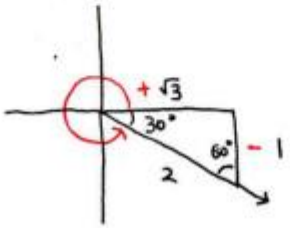
$\csc\left(\frac{-13\pi}{6}\right)$

$\frac{13\pi}{6} \times \frac{180°}{\pi} = 390 = \csc(390°) = \csc(-30°)$

-390
$+360$
-30



$\csc(330°) = \frac{1}{\sin(330°)} = \frac{1}{-\frac{1}{2}} \frac{(-2)}{(-2)} = -2$



$\sin^2\theta + \cos^2\theta = 1$

$1 + \tan^2\theta = \sec^2\theta$

$1 + \cot^2\theta = \csc^2\theta$

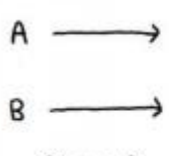
$\sin = \frac{1}{\text{cosec}}$

$\cos = \frac{1}{\sec}$

$\tan = \frac{1}{\cot}$

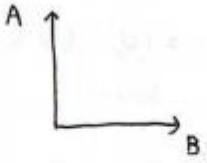
1. 벡터

Parallel



$\theta = 0°$

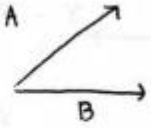
orthogonal



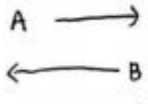
$\theta = 90°$

$\pi/2$

Neither



Perpendicular



$\theta = 180°$

$A = \langle 4, 3 \rangle \quad B = \langle 12, 9 \rangle$

$$\theta = \cos^{-1} \frac{A \cdot B}{\|A\| \cdot \|B\|}$$

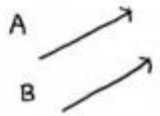
$A \cdot B = 4(12) + 3(9) = 75$

$\|A\| = \sqrt{4^2 + 3^2} = 5$

$\|B\| = \sqrt{12^2 + 9^2} = 15$

$$\theta = \cos^{-1} \frac{75}{5 \cdot 15} = \cos^{-1}(1) = 0°$$

서로 평행



$A = \langle 5, -6 \rangle \quad B = \langle 3, -4 \rangle$

$A \cdot B = 15 + 24 = 39$

$\|A\| = \sqrt{5^2 + (-6)^2} = \sqrt{61}$

$\|B\| = \sqrt{3^2 + (-4)^2} = 5$

Neither

$$\theta = \cos^{-1} \frac{39}{\sqrt{61} \cdot 5} = 2.936°$$

1. 벡터 (내적)

$a = 3i + 4j \quad b = 5i - 2j$

$a = a_x i + a_y j \quad b = b_x i + b_y j$

$a \cdot b = a_x b_x + a_y b_y = 3(5) + 4(-2) = 7$

$a \langle 4, -7 \rangle \quad b \langle -2, 3 \rangle$

$a_x \; a_y \qquad b_x \; b_y$

$a \cdot b = 4(-2) + -7(3) = -29$

$a = 5i - 4j + 3k \quad b = 7i + 2j - 8k$

$a \cdot b = 5(7) + (-4)(2) + 3(-8) = 3$

$a = 5i - 4j \quad b = 7i + 8j \quad c = 3i - 2j$

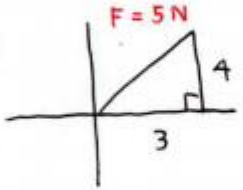
$a \cdot (b + c)$

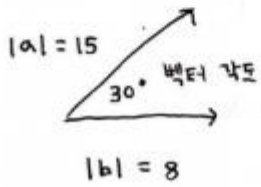
$b + c = 7i + 8j + 3i - 2j = 10i + 6j$

$5(10) + (-4)(6) = 26$

$a = 3i + 4j$

$|a| = \sqrt{a_x^2 + a_y^2} = \sqrt{9 + 16} = 5$





$a \cdot b = |a||b| \cos\theta$

$= 15(8) \cos 30$

$= 120\left(\frac{\sqrt{3}}{2}\right)$

$= 103.92$ (내적)

1. 벡터 (외적)

$a = 3i + 5j - 7k$

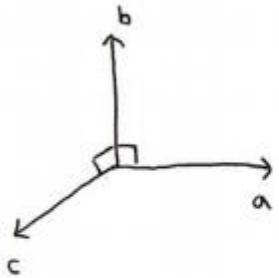
$b = 2i - 6j + 4k$

$$a \times b = \begin{bmatrix} 3 & 5 & -7 \\ 2 & -6 & 4 \end{bmatrix}$$

$$i\begin{bmatrix} 5 & -7 \\ -6 & 4 \end{bmatrix} - j\begin{bmatrix} 3 & -7 \\ 2 & 4 \end{bmatrix} + k\begin{bmatrix} 3 & 5 \\ 2 & -6 \end{bmatrix}$$

$= i(20-42) - j(12--14) + k(-18-10)$

$\vec{c} = \vec{a} \times \vec{b} = -22i - 26j - 28k$



$a \cdot b = 0$ 직교 = 수직

$a \cdot c = 0$ 두 벡터 사이의 각도가

$b \cdot c = 0$ 90° 를 이루는 것

$a = 3i + 5j - 7k$

$b = 2i - 6j + 4k$

$c = -22i - 26j - 28k$

$a \cdot c = -66 - 130 + 196 = 0$

$b \cdot c = -44 + 156 - 112 = 0$

$a = \langle 5, -4, 3 \rangle$ $\vec{a} \times \vec{b} = \vec{c}$

$b = \langle -7, 2, -8 \rangle$

$$\begin{vmatrix} i & j & k \\ 5 & -4 & 3 \\ -7 & 2 & -8 \end{vmatrix} = i\begin{bmatrix} -4 & 3 \\ 2 & -8 \end{bmatrix} - j\begin{bmatrix} 5 & 3 \\ -7 & 8 \end{bmatrix} + k\begin{bmatrix} 5 & -4 \\ -7 & 2 \end{bmatrix}$$

$i(32-6) - j(-40+21) + k(10-28)$

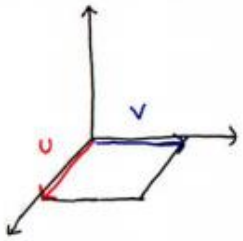
$\vec{c} = 26i + 19j - 18k$

$a \cdot c = 130 - 76 - 54 = 0$

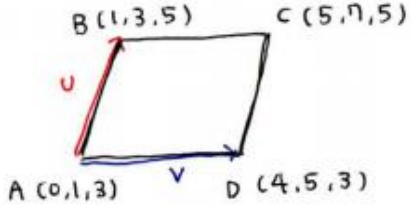
1. 벡터 (평행사변형, 평행육면체)

A (0,1,3), B(1,3,5)

C (5,7,5) D (4,5,3)



Area = $\|U \times V\|$



$U = \overrightarrow{AB} = 1i + 2j + 2k$

$V = \overrightarrow{AD} = 4i + 4j + 0k$

$U \times V = \begin{bmatrix} 1 & 2 & 2 \\ 4 & 4 & 0 \end{bmatrix}$

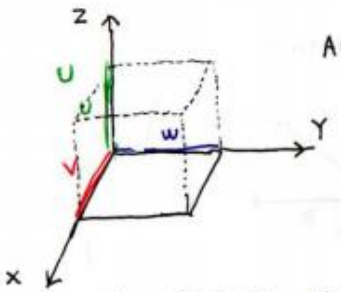
$= i\begin{bmatrix} 2 & 2 \\ 4 & 0 \end{bmatrix} - j\begin{bmatrix} 1 & 2 \\ 4 & 0 \end{bmatrix} + k\begin{bmatrix} 1 & 2 \\ 4 & 4 \end{bmatrix}$

$i(0-8) - j(0-8) + k(4-8)$

$U \times V = -8i + 8j - 4k$

$\|U \times V\| = \sqrt{(-8)^2 + 8^2 + (-4)^2} = \sqrt{144}$

평행사변형 면적 12



Area = $\|V \times W\|$

Volume = $U \cdot (V \times W)$

heigh

$V = Ah$

$V = \langle 5,1,2 \rangle$, $W = \langle 1,4,3 \rangle$, $U = \langle 2,3,6 \rangle$

$$U \cdot (V \times W) = \begin{matrix} U_1 & U_2 & U_3 \\ V_1 & V_2 & V_3 \\ W_1 & W_2 & W_3 \end{matrix} = \begin{matrix} 2 & 3 & 6 \\ 5 & 1 & 2 \\ 1 & 4 & 3 \end{matrix}$$

$= 2\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} - 3\begin{bmatrix} 5 & 2 \\ 1 & 3 \end{bmatrix} + 6\begin{bmatrix} 5 & 1 \\ 1 & 4 \end{bmatrix}$

$= 2(3-8) - 3(15-2) + 6(20-1) = 65$

1. 벡터

수학 이론

$$\begin{matrix} i & j & k \end{matrix}$$
$$\begin{bmatrix} 2 & 3 & 2 \\ 6 & 6 & 6 \end{bmatrix}$$

$$i\begin{bmatrix} 3 & 2 \\ 6 & 6 \end{bmatrix} - j\begin{bmatrix} 2 & 2 \\ 6 & 6 \end{bmatrix} + k\begin{bmatrix} 2 & 3 \\ 6 & 6 \end{bmatrix}$$

$$i(18-12) - j(12-12) + k(12-18)$$

answer : 6i + 0j -6k

C program

```
else
{
    *i = vec_A[1] * vec_B[2] - vec_A[2] * vec_B[1];
    *j = vec_A[0] * vec_B[2] - vec_A[2] * vec_B[0];
    *k = vec_A[0] * vec_B[1] - vec_A[1] * vec_B[0];
}

printf("cross Product:\n");
printf("크기와 방향을 나타내는 벡터 : i , j , k \n");
printf("%di %dj %dk", i, -j, k );
```

```
vector A:
  2  3  2
vector B:
  6  6  6
vector A + B:
  8  9  8
Inner Product:
두 개의 벡터가 서로 수직한가 ?
res = 0.000000
res = 42.000000
cross Product:
크기와 방향을 나타내는 벡터: i , j , k
6i 0j -6k
--------------------------------
```

2. 행렬

Matrix

디지털 비디오를 디코딩하려면 행렬 곱셈이 필요하다
행렬의 행은 열은 픽셀에 해당되고 숫자 항목은 픽셀의 색상 값에 해당한다
Encoding (부호화) : 내가 작성한 코드를 컴퓨터가 사용하는 0 과 1로 변환하는 과정
Byte 형식으로 변환
Decoding (복호화): 사람이 이해할 수 있도록 변경 , 문자열로 변환
Pixel : 이미지를 이루는 가장 작은 단위
모니터 해상도 1024 x 768 , 가로 1024개 , 세로 768개의 픽셀로 이루어진 이미지
총 픽셀 개수 1024 x 768 = 786,432
픽셀은 1 Bit 로 구성되어 있어 1 = 밝음 , 0 = 어두움 (픽셀의 밝기 값 : 1 Byte , 0 ~ 255)
디지털 오디오 신호의 푸리에 변환과 같은 디지털 신호를 필터링 하거나 압축하는 많은 기술들이 행렬 곱셈에 의존한다.

OPENCV ... 이미지 RGB 분리

2. 행렬

$$A = \begin{bmatrix} 2 & 7 & -4 \\ 6 & 3 & 5 \end{bmatrix}$$

2 rows 3 columns

2 × 3

R × C

$$B = \begin{bmatrix} 4 & 3 & 7 & -2 \\ 5 & 6 & -4 & 9 \\ -3 & 8 & 1 & -7 \end{bmatrix}$$

3 × 4

$B_{11} = 4$ $B_{23} = -4$

$B_{14} = -2$ $B_{34} = -7$

$$A = \begin{bmatrix} 3 & -6 \\ 5 & 2 \end{bmatrix} \quad B = \begin{bmatrix} 7 & 4 \\ -5 & 8 \end{bmatrix}$$

$3A + X = B$

X : 행렬

$3A + X = 2B$

$-3A \quad -3A$

$X = 2B - 3A$

$$2\begin{bmatrix} 7 & 4 \\ -5 & 8 \end{bmatrix} \quad -3\begin{bmatrix} 3 & -6 \\ 5 & 2 \end{bmatrix}$$

$$X = \begin{bmatrix} 5 & 26 \\ -25 & 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & 1 & 4 \end{bmatrix} \quad B = \begin{bmatrix} 4 & 3 \\ 2 & 5 \\ 6 & 8 \end{bmatrix}$$

① × ③ ③ × ②

A · B ○

B · A ×

$3(4) + 1(2) + 4(6) = 38$

$9 + 5 + 32 = 46$

$A \cdot B = [\,38 \quad 46\,]$

$$A = \begin{bmatrix} 3 & 4 \\ 7 & 2 \\ 5 & 9 \end{bmatrix} \quad B = \begin{bmatrix} 3 & 1 & 5 \\ 6 & 9 & 7 \end{bmatrix}$$

$A \cdot B = [\,3 \times 2\,] \cdot [\,2 \times 3\,] = 3 \times 3$

$$A \cdot B \begin{bmatrix} 33 & 39 & 43 \\ 33 & 25 & 49 \\ 69 & 86 & 88 \end{bmatrix}$$

2. 행렬

$$A = \begin{bmatrix} 1 & 4 & -2 \\ 3 & 5 & -6 \end{bmatrix} \quad B = \begin{bmatrix} 5 & 2 & 8 & -1 \\ 3 & 6 & 4 & 5 \\ -2 & 9 & 7 & -3 \end{bmatrix}$$

$AB = 2 \times \underset{\uparrow}{3} \quad \underset{\uparrow}{3} \times 4 \rightarrow 2 \times 4$ (A, B; 같다)

$$AB = \begin{bmatrix} 21 & 8 & 10 & 25 \\ 42 & -18 & 2 & 40 \end{bmatrix}$$

$1(5) + 4(3) + (-2)(-2) = 21$

$3(-1) + 5(5) + (-6)(-3) = 40$

$$\begin{aligned} 2x - 7y &= 1 \\ 3x + y &= 13 \end{aligned}$$

$$D = \begin{bmatrix} 2 & -7 \\ 3 & 1 \end{bmatrix} = 2(1) - 3(-7) = 23$$

$$D_x = \begin{bmatrix} 1 & -7 \\ 13 & 1 \end{bmatrix} = 1 - 13(-7) = 92$$

$$D_y = \begin{bmatrix} 2 & 1 \\ 3 & 13 \end{bmatrix} = 2(13) - 3(1) = 23$$

$$x = \frac{D_x}{D} = \frac{92}{23} = 4$$

$$y = \frac{D_y}{D} = \frac{23}{23} = 1$$

(4, 1)

$$\begin{aligned} 3x - 2y + 1z &= 2 \\ 4x + 3y - 2z &= 4 \\ 5x - 3y + 3z &= 8 \end{aligned}$$

$D_y = 52$

$D_z = 78$

(1, 2, 3)

$$D = \begin{bmatrix} 3 & -2 & 1 \\ 4 & 3 & -2 \\ 5 & -3 & 3 \end{bmatrix} \quad D_x = \begin{bmatrix} 2 & -2 & 1 \\ 4 & 3 & -2 \\ 8 & -3 & 3 \end{bmatrix}$$

$$3\begin{bmatrix} 3 & -2 \\ -3 & 3 \end{bmatrix} - (-2)\begin{bmatrix} 4 & -2 \\ 5 & 3 \end{bmatrix} + 1\begin{bmatrix} 4 & 3 \\ 5 & -3 \end{bmatrix} \quad D = 26$$

(9 − 6, 12 − −10, −12 − 15)

$$2\begin{bmatrix} 3 & -2 \\ -3 & 3 \end{bmatrix} - (-2)\begin{bmatrix} 4 & -2 \\ 8 & 3 \end{bmatrix} + 1\begin{bmatrix} 4 & 3 \\ 8 & -3 \end{bmatrix} \quad D_x = 26$$

2. 행렬

```
printf("\n");

for(i = 0; i < arows; i++)
{
    for(j = 0; j < bcolumns; j++)
    {
        for(k = 0; k < brows; k++)
        {
            sum += a[i][k] * b[k][j];
        }
        product[i][j] = sum;
        sum = 0;
    }
}

printf("행 렬  결 과  \n");

for(i = 0; i < arows; i++)
{
    for(j = 0; j < bcolumns; j++)
    {
        printf("%3d", product[i][j]);
    }
    printf("\n");

}
```

```
 a의 행과 열 입력 rows x columns :
3 2
 a 행렬 요소들 값 입력
3 4
7 2
5 9
 b의 행과 열 입력 rows x columns :
2 3
b 행렬 요소들 값 입력:
3 1 5
6 9 7

행렬 결과
 33 39 43
 33 25 49
 69 86 88
```

2. 역행렬

$$\begin{bmatrix} 1 & 0 & 4 & -6 \\ 2 & 5 & 0 & 3 \\ -1 & 2 & 3 & 5 \\ 2 & 1 & -2 & 3 \end{bmatrix}$$

$$74 - 0 + 4(52) + 6(6) = 318$$

$$+1\begin{bmatrix} 5 & 0 & 3 \\ 2 & 3 & 5 \\ 1 & -2 & 3 \end{bmatrix} - 0\begin{bmatrix} 2 & 0 & 3 \\ -1 & 3 & 5 \\ 2 & -2 & 3 \end{bmatrix} + 4\begin{bmatrix} 2 & 5 & 3 \\ -1 & 2 & 5 \\ 2 & 1 & 3 \end{bmatrix} - (-6)\begin{bmatrix} 2 & 5 & 0 \\ -1 & 2 & 3 \\ 2 & 1 & -2 \end{bmatrix}$$

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad A^{-1} = \frac{1}{ad - bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

$$A = \begin{bmatrix} 7 & 2 \\ 17 & 5 \end{bmatrix} \quad A^{-1} = \frac{1}{35 - 34}\begin{bmatrix} 5 & -2 \\ -17 & 7 \end{bmatrix}$$

$$A \cdot A^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 7 & 2 \\ 17 & 5 \end{bmatrix}\begin{bmatrix} 5 & -2 \\ -17 & 7 \end{bmatrix}$$

$$\uparrow A^{-1}$$

$$\begin{bmatrix} 7(5) + 2(-17) & 7(-2) + 2(7) \\ 17(5) + 5(-17) & 17(-2) + 5(7) \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 2 & -1 \\ -2 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix} \quad A^{-1} \quad [A \mid I_3]$$

$$\left[\begin{array}{ccc|ccc} 1 & 2 & -1 & 1 & 0 & 0 \\ -2 & 0 & 1 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 1 \end{array}\right] \xrightarrow{R_1 - R_3} \left[\begin{array}{ccc|ccc} 1 & 2 & -1 & 1 & 0 & 0 \\ -2 & 0 & 1 & 0 & 1 & 0 \\ 0 & 3 & -1 & 1 & 0 & -1 \end{array}\right]$$

$$\xrightarrow{2R_1 + R_2} \left[\begin{array}{ccc|ccc} 1 & 2 & -1 & 1 & 0 & 0 \\ 0 & 4 & -1 & 2 & 1 & 0 \\ 0 & 3 & -1 & 1 & 6 & -1 \end{array}\right] \xrightarrow{3R_2 - 4R_3}$$

$$\left[\begin{array}{ccc|ccc} 1 & 2 & -1 & 1 & 0 & 0 \\ 0 & 4 & -1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 3 & 4 \end{array}\right] \xrightarrow{R_1 + R_3} \left[\begin{array}{ccc|ccc} 1 & 2 & 0 & 3 & 3 & 4 \\ 0 & 4 & -1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 3 & 4 \end{array}\right]$$

$$\xrightarrow{R_2 + R_3} \left[\begin{array}{ccc|ccc} 1 & 2 & 0 & 3 & 3 & 4 \\ 0 & 4 & 0 & 4 & 4 & 4 \\ 0 & 0 & 1 & 2 & 3 & 4 \end{array}\right] \xrightarrow{2R_1 - R_2} \left[\begin{array}{ccc|ccc} 2 & 0 & 0 & 2 & 2 & 4 \\ 0 & 4 & 0 & 4 & 4 & 4 \\ 0 & 0 & 1 & 2 & 3 & 4 \end{array}\right]$$

$$\xrightarrow{\frac{1}{2}R_1} \xrightarrow{\frac{1}{4}R_2} \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & 1 & 2 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 2 & 3 & 4 \end{array}\right]$$

$$\uparrow I_3 \qquad \uparrow A^{-1}$$

2. 역행렬

$$x + y - z = -2$$
$$2x - y + z = 5$$
$$-x + 2y + 2z = 1$$

$$\left[\begin{array}{ccc|c} 1 & 1 & -1 & -2 \\ 2 & -1 & 1 & 5 \\ -1 & 2 & 2 & 1 \end{array}\right] \xrightarrow{R_1 + R_3}$$

1이거나 0

$$\left[\begin{array}{ccc|c} 1 & 1 & -1 & -2 \\ 2 & -1 & 1 & 5 \\ 0 & 3 & 1 & 1 \end{array}\right] \xrightarrow{-2R_1 + R_2} \left[\begin{array}{ccc|c} 1 & 1 & -1 & -2 \\ 0 & -3 & 3 & 9 \\ 0 & 3 & 1 & -1 \end{array}\right] \xrightarrow{R_2 + R_3}$$

$$\left[\begin{array}{ccc|c} 1 & 1 & -1 & -2 \\ 0 & -3 & 3 & 9 \\ 0 & 0 & 4 & 8 \end{array}\right] \xrightarrow[1/4\,R_3]{-1/3\,R_2} \left[\begin{array}{ccc|c} 1 & 1 & -1 & 2 \\ 0 & 1 & -1 & -3 \\ 0 & 0 & 1 & 2 \end{array}\right]$$

$$x + y - z = 2$$
$$y - z = -3$$
$$z = 2$$

$$y - 2 = -3$$
$$+2 \quad +2$$
$$y = -1$$

$$x + (-1) - (2) = -2$$
$$x - 3 = -2$$
$$+3 \quad +3$$
$$x = 1$$

2. 역행렬 (보류....)

```
printf("행 렬  요 소 들  값  입 력  :\n");
for(i=0;i<sizeOfMatrix;i++)
    for(j=0;j<sizeOfMatrix;j++)
        scanf("%f",&input[i][j]);

for(i=0;i<sizeOfMatrix;i++)
for(j=0;j<sizeOfMatrix;j++)
if(i==j)
Inverse[i][j]=1;
else
Inverse[i][j]=0;
 printf("역 행 렬  결 과 :\n");

 for(i=0;i<sizeOfMatrix;i++)
 {
     for(j=0;j<sizeOfMatrix;j++)
         printf("%f  ",Inverse[i][j]);
     printf("\n");
 }
 return 0;
```

```
행렬 정사각형 입력 n x n
2
행렬 요소들 값 입력 :
7 2
17 5
역행렬 결:
5.000002        -2.000001
-17.000008      7.000003
```

2. 구조체

C program

```
struct point {
int x, y;
};

struct rect {
struct point p1;
struct point p2;
float a, b, c;
};

int main(void)
{
struct rect r ;
struct rect p = { .b = 10.11,  .c = 12.33,  .a = 14.33};
int w, h;
scanf("%d %d", &r.p1.x, &r.p1.y);
scanf("%d %d", &r.p2.x, &r.p2.y);
w = r.p2.x - r.p1.x;
h = r.p2.y - r.p1.y;
printf("w 는 %d , h는 %d₩n", w, h);
printf("a : %f , b : %f , c : %f", p.a, p.b, p.c);
```

2. 구조체

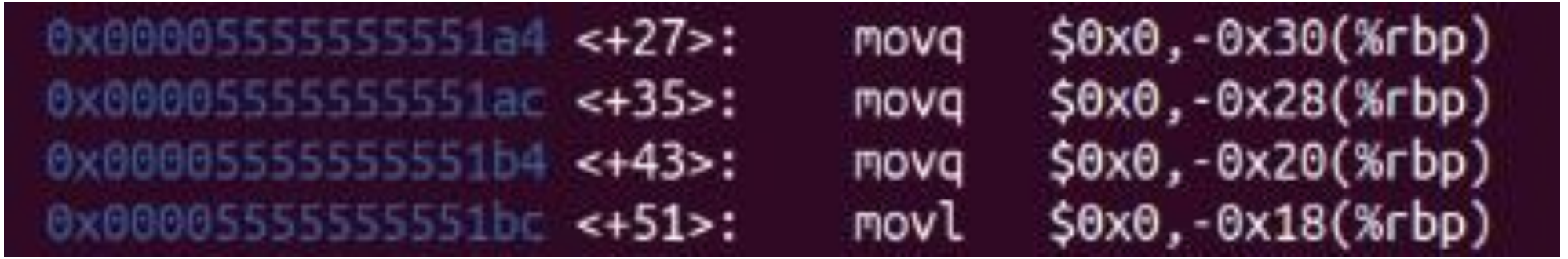
assembly

지역변수 초기 값

```
r = {p1 = {x = 194, y = 0}, p2 = {x = -8665, y = 32767},
      a = -nan(0x7fde26), b = 4.59163468e-41, c = 1.46595255e+13}
p = {p1 = {x = -134557752, y = 32767}, p2 = {x = 1431655088, y = 21845},
      a = 0, b = 0, c = 1.46588912e+13}
w = 15775231
h = 0
```

<+27>, <+35>, <+43> <+51> 실행

```
p = {p1 = {x = 0, y = 0}, p2 = {x = 0, y = 0}, a = 0, b = 0, c = 0}
```



2. 구조체

assembly

<+66>, <+79>, <+92> 실행

p = {p1 = {x = 0, y = 0}, p2 = {x = 0, y = 0}, a = 14.3299999, b = 10.1099997, c = 12.3299999}

첫 번째 scanf 1, 1 입력 / 두 번째 scanf 6,6 입력

r = {p1 = {x = 1, y = 1}, p2 = {x = 6, y = 6},

```
0x00005555555551c3 <+58>:    movss  0xe6d(%rip),%xmm0        # 0x555555556038
0x00005555555551cb <+66>:    movss  %xmm0,-0x20(%rbp)
0x00005555555551d0 <+71>:    movss  0xe64(%rip),%xmm0        # 0x55555555603c
0x00005555555551d8 <+79>:    movss  %xmm0,-0x1c(%rbp)
0x00005555555551dd <+84>:    movss  0xe5b(%rip),%xmm0        # 0x555555556040
0x00005555555551e5 <+92>:    movss  %xmm0,-0x18(%rbp)
```

2. 구조체

assembly

movss (Move scalar single-precision floating-point values)

: 부동 소수점 값 하나를 복사해주는 명령어

%xmm0 ~ 15 (128 Bit)

: packed/scalar 단정밀 FP 연산자를 XMM 레지스터와 메모리에 전송

cvtss2sd (Convert Scalar Single-Precision Floating-Point Value to Scalar Double-Precision Floating-Point Value)

: xmm2 / m32에있는 하나의 단 정밀도 부동 소수점 값을 xmm1에 있는 하나의 배정 부동 소수점 값으로 변환

```
   0x0000555555555261 <+216>:   movss  -0x18(%rbp),%xmm0
   0x0000555555555266 <+221>:   cvtss2sd %xmm0,%xmm2
   0x000055555555526a <+225>:   movss  -0x1c(%rbp),%xmm0
   0x000055555555526f <+230>:   cvtss2sd %xmm0,%xmm1
   0x0000555555555273 <+234>:   movss  -0x20(%rbp),%xmm0
--Type <RET> for more, q to quit, c to continue without paging--
   0x0000555555555278 <+239>:   cvtss2sd %xmm0,%xmm0
```

3. 배열 포인터

C program

```
int main()
{

int i;
int a[][3] = {1, 2, 3, 4, 5, 6};
int (*ptr1)[3] = a;

printf("%d, %d₩n", (*ptr1)[1], (*ptr1)[2]);

++ptr1;

printf("%d, %d₩n", (*ptr1)[1], (*ptr1)[2]);

return 0;
}
```

기타

*(a+0) = a [0] = &a[0][0]

**a = 1

(*ptr)[1] = *((*ptr) + 1)

*((*ptr) + 0) = 1
*((*ptr) + 1) = 2
*((*ptr) + 2) = 3

*((*ptr+1) + 0) = 4
*((*ptr+1) + 1) = 5
*((*ptr+1) + 2) = 6

3. 배열 포인터

```
2.c
  1         #include <stdio.h>
  2         #include <stdlib.h>
  3
  4         int main()
  5
  6         {
  7
  8                 int i;
  9                 int a[][3] = {1, 2, 3, 4, 5, 6};
 10                 int (*ptr1)[3] = a;
 11
 12                 printf("%d, %d\n", (*ptr1)[1], (*ptr1)[2]);
B+>13               ++ptr1;
 14                 printf("%d, %d\n", (*ptr1)[1], (*ptr1)[2]);
 15
 16                 return 0;
 17         }
```

```
native process 3237 In: main                                                    5551d7
Dump of assembler code for function main:
   0x00005555555551a7 <+62>:    movl   $0x6,-0xc(%rbp)
   0x00005555555551ae <+69>:    lea    -0x20(%rbp),%rax
   0x00005555555551b2 <+73>:    mov    %rax,-0x28(%rbp)
   0x00005555555551b6 <+77>:    mov    -0x28(%rbp),%rax
   0x00005555555551ba <+81>:    mov    0x8(%rax),%edx
   0x00005555555551bd <+84>:    mov    -0x28(%rbp),%rax
   0x00005555555551c1 <+88>:    mov    0x4(%rax),%eax
   0x00005555555551c4 <+91>:    mov    %eax,%esi
   0x00005555555551c6 <+93>:    lea    0xe37(%rip),%rdi        # 0x555555556004
   0x00005555555551cd <+100>:   mov    $0x0,%eax
   0x00005555555551d2 <+105>:   callq  0x555555555070 <printf@plt>
=> 0x00005555555551d7 <+110>:   addq   $0xc,-0x28(%rbp)
   0x00005555555551dc <+115>:   mov    -0x28(%rbp),%rax
--Type <RET> for more, q to quit, c to continue without paging--
```

메모리 주소 값
56864 : 1 그룹 대표 주소
56876 : 2 그룹 대표 주소

++ptr	실행 전	실행 후
56896	rbp	rbp
56884	6	6
56880	5	5
56876	4	4
56872	3	3
56868	2	2
56864	1	1
56856	1 그룹	2그룹

End of Document