EDDI
Electronic Design
Development Institute

에디로봇아카데미

임베디드 마스터 Lv2 과정

제 1기

2022. 04. 02

손표훈

CONTENTS

행렬 덧셈,뺄셈

EDDI Electronic Design Development Institute

→ 행렬의 덧셈, 뺄셈은 같은 꼴의 행렬만 덧셈 뺄셈이 가능하다
예) 2x2는 2x2끼리 3x3은 3x3끼리
→ 행렬의 덧셈, 뺄셈은 같은 행,열의 성분끼리 더하고 뺀다
→ 행렬의 덧셈, 뺄셈은 교환법칙(A+B = B+A), 결합법칙(A+(B+C) = (A+B)+C)

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

$$A + B = \begin{bmatrix} 2 & 3 & 1 \\ 5 & 2 & 1 \\ 1 & 6 & 7 \end{bmatrix} + \begin{bmatrix} 4 & 1 & 6 \\ 5 & 1 & 7 \\ 1 & 3 & 9 \end{bmatrix} = \begin{bmatrix} 2+4 & 3+1 & 1+6 \\ 5+5 & 2+1 & 1+7 \\ 1+1 & 6+3 & 7+9 \end{bmatrix} = \begin{bmatrix} 6 & 4 & 7 \\ 10 & 3 & 8 \\ 2 & 9 & 16 \end{bmatrix}$$

$$A - B = \begin{bmatrix} 2 & 3 & 1 \\ 5 & 2 & 1 \\ 1 & 6 & 7 \end{bmatrix} - \begin{bmatrix} 4 & 1 & 6 \\ 5 & 1 & 7 \\ 1 & 3 & 9 \end{bmatrix} = \begin{bmatrix} 2-4 & 3-1 & 1-6 \\ 5-5 & 2-1 & 1-7 \\ 1-1 & 6-3 & 7-9 \end{bmatrix} = \begin{bmatrix} -2 & 2 & -5 \\ 0 & 1 & -6 \\ 0 & 3 & -2 \end{bmatrix}$$

행렬 곱셈

→ 두개의 행렬의 곱셈은 열=행 같은 꼴만 가능하다
예) A = pxq, B = rxs 꼴의 행렬이라면, q=r일 때 곱셈이 가능하다
→ 각 행렬의 행x열 성분끼리 곱하고 더한 결과 pxs꼴의 행렬이 결과가 된다
→ 행렬의 곱셈은 교환법칙이 성립하지 않는다

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

$$A \times B = \begin{bmatrix} 2 & 3 & 1 \\ 5 & 2 & 1 \\ 1 & 6 & 7 \end{bmatrix} \times \begin{bmatrix} 4 & 1 & 6 \\ 5 & 1 & 7 \\ 1 & 3 & 9 \end{bmatrix} = \begin{bmatrix} 8+15+1 & 2+3+3 & 6+7+9 \\ 20+10+1 & 5+2+3 & 30+14+9 \\ 4+30+7 & 1+6+21 & 6+42+63 \end{bmatrix} = \begin{bmatrix} 24 & 8 & 22 \\ 31 & 10 & 53 \\ 41 & 28 & 111 \end{bmatrix}$$

$$B \times A = \begin{bmatrix} 4 & 1 & 6 \\ 5 & 1 & 7 \\ 1 & 3 & 9 \end{bmatrix} \times \begin{bmatrix} 2 & 3 & 1 \\ 5 & 2 & 1 \\ 1 & 6 & 7 \end{bmatrix} = \begin{bmatrix} 8+5+6 & 12+2+36 & 4+1+42 \\ 10+25+7 & 3+2+6 & 7+7+49 \\ 2+15+1 & 3+6+54 & 1+3+63 \end{bmatrix} = \begin{bmatrix} 19 & 50 & 47 \\ 42 & 11 & 63 \\ 18 & 63 & 67 \end{bmatrix}$$

$\therefore A \times B \neq B \times A$ 교환법칙이 성립하지 않는다

행렬의 스케일링

→ 행렬의 스케일링(스칼라 곱)은 행렬의 각 성분에 상수를 곱한다

$$A = \begin{bmatrix} 2 & 3 & 1 \\ 5 & 2 & 1 \\ 1 & 6 & 7 \end{bmatrix}, k = 2$$

$$kA = 2\begin{bmatrix} 2 & 3 & 1 \\ 5 & 2 & 1 \\ 1 & 6 & 7 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 2 \\ 10 & 4 & 2 \\ 2 & 12 & 14 \end{bmatrix}$$

역행렬(행렬식, 수반행렬, 전치행렬)

→ 주어진 nxn행렬(정방행렬) A,C를 알고 있을 면, AB = C에서 B를 구하기 위해 어떤 행렬 X를 곱했을 때 XAB = XC, XA는 단위 행렬이 되면 B행렬을 구 할 수 있다. X = A^{-1}

→ 단위 행렬 I는 주대각 성분이 모두 1이고 그 외 성분은 0인 행렬이다

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

→ 수반행렬(adjoint matrix)과 행렬식, 전치행렬을 이용한 2x2 행렬의 역행렬

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

(1)$\det(A) = a_{11}\ a_{22} - a_{12}a_{21}$

(2)전치행렬 $A^T = \begin{bmatrix} a_{11} & a_{21(12)} \\ a_{12(21)} & a_{22} \end{bmatrix}$

(3)M_{ij} = 소행렬로 A의 전치 행렬에서 i,j의 행과 열을 제외한 성분의 행렬을 의미한다

$M_{11} = a_{22}$

$M_{12} = a_{21}$

$M_{21} = a_{12}$

$M_{22} = a_{11}$

(4)A의 여인수 = $C_{ij} = (-1)^{i+j}\det(M_{ij})$

$C_{11} = (-1)^{1+1}\det(M_{11}) = a_{22}$

$C_{12} = (-1)^{1+2}\det(M_{12}) = -a_{12}$

$C_{21} = (-1)^{2+1}\det(M_{21}) = -a_{21}$

$C_{22} = (-1)^{2+2}\det(M_{22}) = a_{11}$

$$\therefore adj(A) = \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}$$

(5)$A^{-1} = \dfrac{1}{\det(A)} adj(A) = \dfrac{1}{a_{11}\ a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}$

→ 수반행렬은 A의 여인수로 이루어진 행렬(C)이다. 여인수 행렬은 다음과 같은 성질이 있다 A^T의 C = A의 C

→ det(A)가 0일 때 역행렬이 존재 하지 않는다(peudo inverse matrix를 사용하여 구할 수 있다)

역행렬(행렬식, 수반행렬, 전치행렬)

→ 수반행렬(adjoint matrix)과 행렬식, 전치행렬을 이용한 3x3 행렬의 역행렬을 구하기

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$(1)\det(A) = a_{11}\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12}\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} = a_{11}(a_{22}\ a_{33} \text{-}\ a_{23}\ a_{32})\text{-}a_{12}(a_{21}\ a_{33} \text{-}\ a_{23}\ a_{31})\text{-}a_{13}(a_{21}\ a_{32} \text{-}\ a_{22}\ a_{31})$$

$$(2)A^T = \begin{bmatrix} a_{11} & a_{21(12)} & a_{31(13)} \\ a_{12(21)} & a_{22} & a_{32(23)} \\ a_{13(31)} & a_{23(32)} & a_{33} \end{bmatrix}$$

$$(3)\ M_{11} = \begin{bmatrix} a_{22} & a_{32} \\ a_{23} & a_{33} \end{bmatrix}, M_{12} = \begin{bmatrix} a_{12} & a_{32} \\ a_{13} & a_{33} \end{bmatrix}, M_{13} = \begin{bmatrix} a_{12} & a_{22} \\ a_{13} & a_{23} \end{bmatrix}, M_{21} = \begin{bmatrix} a_{21} & a_{31} \\ a_{23} & a_{33} \end{bmatrix}, M_{22} = \begin{bmatrix} a_{11} & a_{31} \\ a_{13} & a_{33} \end{bmatrix}, M_{23} = \begin{bmatrix} a_{11} & a_{21} \\ a_{13} & a_{23} \end{bmatrix}$$

$$M_{31} = \begin{bmatrix} a_{21} & a_{31} \\ a_{22} & a_{32} \end{bmatrix}, M_{32} = \begin{bmatrix} a_{11} & a_{31} \\ a_{12} & a_{32} \end{bmatrix}, M_{33} = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{bmatrix}$$

$$(4)\ adj(A) = \begin{bmatrix} \det(M_{11}) & \det(M_{12}) & \det(M_{13}) \\ \det(M_{21}) & \det(M_{22}) & \det(M_{23}) \\ \det(M_{31}) & \det(M_{32}) & \det(M_{33}) \end{bmatrix} = \begin{bmatrix} a_{22}a_{33} - a_{32}a_{23} & -(a_{12}a_{33} - a_{32}a_{13}) & a_{12}a_{23} - a_{22}a_{13} \\ -(a_{21}a_{33} - a_{31}a_{23}) & a_{11}a_{33} - a_{31}a_{13} & -(a_{11}a_{23} - a_{21}a_{13}) \\ a_{21}a_{32} - a_{31}a_{22} & -(a_{11}a_{32} - a_{31}a_{12}) & a_{11}a_{22} - a_{21}a_{12} \end{bmatrix}$$

$$(5)\ A^{-1} = \frac{1}{\det(A)}adj(A) = \begin{bmatrix} \det(M_{11}) & \det(M_{12}) & \det(M_{13}) \\ \det(M_{21}) & \det(M_{22}) & \det(M_{23}) \\ \det(M_{31}) & \det(M_{32}) & \det(M_{33}) \end{bmatrix} = \frac{1}{\det(A)}\begin{bmatrix} a_{22}a_{33} - a_{32}a_{23} & -(a_{12}a_{33} - a_{32}a_{13}) & a_{12}a_{23} - a_{22}a_{13} \\ -(a_{21}a_{33} - a_{31}a_{23}) & a_{11}a_{33} - a_{31}a_{13} & -(a_{11}a_{23} - a_{21}a_{13}) \\ a_{21}a_{32} - a_{31}a_{22} & -(a_{11}a_{32} - a_{31}a_{12}) & a_{11}a_{22} - a_{21}a_{12} \end{bmatrix}$$

크래머 공식

→ 선형 연립방정식을 푸는 방법 중 하나

→ 다음과 같은 선형 연립방정식이 있을 때 크래머 공식을 이용하여 해를 구할 수 있다.

$$\begin{aligned} b_1 &= a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n \\ b_2 &= a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n \\ b_3 &= a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n \\ &\vdots \\ b_n &= a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n \end{aligned}$$

→ 위 연립방정식은 다음과 같이 행렬로 표현 할 수 있다.

(1) $A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix}, X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, B = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \rightarrow B = AX \rightarrow \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ X는 다음과 같이 구할 수 있다.

(2) $A^{-1}B = A^{-1}AX \rightarrow A^{-1}B = X \rightarrow X = \frac{adj(A)}{\det(A)}B \rightarrow \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \frac{1}{\det(A)} \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{nn} \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$

(3) $\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \frac{1}{\det(A)} \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{nn} \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$ 따라서 해는 다음과 같다.

$$x_1 = \frac{c_{11}b_1 + c_{12}b_2 + \cdots + c_{1n}b_n}{\det(A)}, x_2 = \frac{c_{21}b_1 + c_{22}b_2 + \cdots + c_{2n}b_n}{\det(A)}, \cdots, x_n = \frac{c_{n1}b_1 + c_{n2}b_2 + \cdots + c_{nn}b_n}{\det(A)}$$

크래머 공식

- 크래머 공식을 이용한 연립방정식의 해

  → 아래와 같은 선형 연립 방정식이 있다

$$\begin{aligned} b_1 &= a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n \\ b_2 &= a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n \\ b_3 &= a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n \end{aligned} \qquad A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$(1)\ x_1 = \frac{\begin{vmatrix} b_1 & a_{12} & a_{13} \\ b_2 & a_{22} & a_{23} \\ b_3 & a_{32} & a_{33} \end{vmatrix}}{\det(A)} = \frac{1}{\det(A)}(b_1((a_{22}a_{33} - a_{23}a_{32}) = c_{11})) + \frac{1}{\det(A)}(b_2((a_{12}a_{33} - a_{13}a_{32}) = c_{21})) + \frac{1}{\det(A)}(b_3((a_{12}a_{23} - a_{13}a_{22}) = c_{31}))$$

$$(2)\ x_2 = \frac{\begin{vmatrix} a_{11} & b_1 & a_{13} \\ a_{21} & b_2 & a_{23} \\ a_{31} & b_3 & a_{33} \end{vmatrix}}{\det(A)} = \frac{1}{\det(A)}(b_1((a_{21}a_{33} - a_{23}a_{31}) = c_{12})) + \frac{1}{\det(A)}(b_2((a_{11}a_{33} - a_{13}a_{31}) = c_{22})) + \frac{1}{\det(A)}(b_3((a_{11}a_{23} - a_{13}a_{21}) = c_{32}))$$

$$(3)\ x_3 = \frac{\begin{vmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ a_{31} & a_{32} & b_3 \end{vmatrix}}{\det(A)} = \frac{1}{\det(A)}(b_1((a_{21}a_{32} - a_{22}a_{31}) = c_{13})) + \frac{1}{\det(A)}(b_2((a_{11}a_{32} - a_{12}a_{31}) = c_{23})) + \frac{1}{\det(A)}(b_3((a_{11}a_{22} - a_{12}a_{21}) = c_{33}))$$

크래머 공식

→ 크래머 공식을 C로 구현하기 위한 전략

(1) 해를 위한 포인터, 계수행렬의 포인터, 결과 행렬의 포인터

(2) 계수행렬의 행렬식 구하기

(3) 크래머 공식을 위한 분자의 행렬 생성

(4) (3)의 인자로 계수행렬, 열 인덱스, 결과 행렬의 포인터

(5) (3)의 행렬의 행렬식을 구한 뒤 결과행렬에 저장

(6) 결과 행렬에 (5)의 결과행렬/det(계수행렬)을 저장한다

```
void molding_mat(float (*A)[3], float *ans, int idx, float (*R)[3])
{
	int i, j;

#if 0
	00		01		02
	10		11		12
	20		21		22
#endif

	for(i = 0; i < 3; i++)
	{
		for(j = 0; j < 3; j++)
		{
			if(j == idx)
				continue;
			R[i][j] = A[i][j];
		}

		R[i][idx] = ans[i];
	}
}
```

```
void crammer_formula(float (*A)[3], float *ans, float *xyz)
{
	float detA, detX, detY, detZ;
	float R[3][3] = {};

	detA = det_mat(A);

	molding_mat(A, ans, 0, R);
#ifdef __DEBUG__
	print_mat(R);
#endif
	detX = det_mat(R);

	molding_mat(A, ans, 1, R);
#ifdef __DEBUG__
	print_mat(R);
#endif
	detY = det_mat(R);

	molding_mat(A, ans, 2, R);
#ifdef __DEBUG__
	print_mat(R);
#endif
	detZ = det_mat(R);

	xyz[0] = detX / detA;
	xyz[1] = detY / detA;
	xyz[2] = detZ / detA;
}
```

가우스 조던 소거법 공식

EDDI Electronic Design Development Institute

➢ 선형 연립방정식의 해 구하기

예 : 다음과 같은 연립방정식이 있다

$$\begin{aligned} x_1 + 3x_2 + x_3 &= 9 \\ x_1 + x_2 - x_3 &= 1 \\ 3x_1 + 11x_2 + 5x_3 &= 35 \end{aligned} \longrightarrow \begin{bmatrix} 1 & 3 & 1 \\ 1 & 1 & -1 \\ 3 & 11 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 9 \\ 1 \\ 35 \end{bmatrix}$$

#1. 해를 포함하여 3x4 첨가행렬을 생성한다

$$\left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 1 & 1 & -1 & 1 \\ 3 & 11 & 5 & 35 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

#2. 주대각 성분은 1이 되게 하고 삼각형/역삼각형 모양으로 만든다

$$\left(\begin{array}{ccc|c} 1 & a & b & v \\ 0 & 1 & c & w \\ 0 & 0 & 1 & z \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad \left(\begin{array}{ccc|c} 1 & 0 & 0 & v \\ a & 1 & 0 & w \\ b & c & 1 & z \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

#3. 1행 1열을 제외한 1열의 성분들을 0으로 만든다 : 2행 - (2행 1열/1행 1열)*1행, 3행 – (3행 1열/1행 1열)*1행을 계산한 뒤 2,3행에 각각 대입한다

(1) $$\left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 1 & 1 & -1 & 1 \\ 3 & 11 & 5 & 35 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \longrightarrow \frac{2\text{행 } 1\text{열}}{1\text{행 } 1\text{열}} = 1 \longrightarrow \left(\begin{array}{ccc|c} 1\times 1 & 3\times 1 & 1\times 1 & 9\times 1 \\ 1 & 1 & -1 & 1 \\ 3 & 11 & 5 & 35 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \xrightarrow{2\text{행-}1\text{행}} \left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 & -2 & -8 \\ 3 & 11 & 5 & 35 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

(2) $$\left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 & -2 & -8 \\ 3 & 11 & 5 & 35 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \longrightarrow \frac{2\text{행 } 1\text{열}}{1\text{행 } 1\text{열}} = 3 \longrightarrow \left(\begin{array}{ccc|c} 1\times 3 & 3\times 3 & 1\times 3 & 9\times 3 \\ 1 & 1 & -1 & -8 \\ 3 & 11 & 5 & 35 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \xrightarrow{3\text{행-}1\text{행}} \left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 & -2 & -8 \\ 0 & 2 & 2 & 8 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

가우스 조던 소거법 공식

#4. 2행 2열을 제외한 2열의 성분들을 0으로 만든다 : 1행 - (1행 2열/2행 2열)*2행, 3행 – (3행 2열/2행 2열)*2행을 계산한 뒤 1,3행에 각각 대입한다

(1) $\left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 & -2 & -8 \\ 0 & 2 & 2 & 8 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \longrightarrow \frac{1\text{행 } 2\text{열}}{2\text{행 } 2\text{열}} = \frac{-3}{2} \longrightarrow \left(\begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 \times \frac{-3}{2} & -2 \times \frac{-3}{2} & -8 \times \frac{-3}{2} \\ 0 & 2 & 2 & 8 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \xrightarrow{1\text{행-}2\text{행}} \left(\begin{array}{ccc|c} 1 & 0 & -2 & -3 \\ 0 & -2 & -2 & -8 \\ 0 & 2 & 2 & 8 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$

(2) $\left(\begin{array}{ccc|c} 1 & 0 & -2 & -3 \\ 0 & -2 & -2 & -8 \\ 0 & 2 & 2 & 8 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \longrightarrow \frac{3\text{행 } 2\text{열}}{2\text{행 } 2\text{열}} = -1 \longrightarrow \left(\begin{array}{ccc|c} 1 & 0 & -2 & -3 \\ 0 & -2 \times -1 & -2 \times -1 & -8 \times -1 \\ 0 & 2 & 2 & 8 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \xrightarrow{3\text{행-}2\text{행}} \left(\begin{array}{ccc|c} 1 & 0 & -2 & -3 \\ 0 & -2 & -2 & -8 \\ 0 & 0 & 0 & 0 \end{array}\right) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$

#5. 3행 3열을 제외한 2열의 성분들을 0으로 만든다 : 1행 - (1행 3열/3행 3열)*1행, 2행 – (2행 3열/3행 3열)*2행을 계산한 뒤 1,3행에 각각 대입한다

* 3행 3열이 0인 경우 계산을 끝낸다

#6. x_3 = 3행 3열 x_2 = (2행 4열-(2행 3열 x x_3))/2행 2열, x_1 = (1행 4열-(1행 2열 x x_2 + 1행 3열 x x_3))/1행 1열

가우스 조던 소거법 공식

➢ 역행렬 구하기

→ 어떤 행렬 X가 A의 역행렬이라면 XA = I가 된다.

*연립 방정식의 해를 구하는 절차와 동일하다

#1 단위행렬을 포함하여 3x6 첨가행렬을 생성한다

$$X = \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ -6 & 49 & -10 & 0 & 1 & 0 \\ -4 & 34 & -5 & 0 & 0 & 1 \end{array}\right)$$

#2 1행 1열을 제외한 1열의 성분들을 0으로 만든다 : 2행 - (2행 1열/1행 1열)*1행, 3행 – (3행 1열/1행 1열)*1행을 계산한 뒤 2,3행에 각각 대입한다

(1) $$X = \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ -6 & 49 & -10 & 0 & 1 & 0 \\ -4 & 34 & -5 & 0 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1\times 6 & 8\times 6 & -2\times 6 & 1\times 6 & 0 & 0 \\ -6 & 49 & -10 & 0 & 1 & 0 \\ -4 & 34 & -5 & 0 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ -4 & 34 & -5 & 0 & 0 & 1 \end{array}\right)$$

(2) $$X = \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ -6 & 49 & -10 & 0 & 1 & 0 \\ -4 & 34 & -5 & 0 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1\times 4 & 8\times 4 & -2\times 4 & 1\times 4 & 0 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ -4 & 34 & -5 & 0 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 2 & 3 & -4 & 0 & 1 \end{array}\right)$$

#3 2행 2열을 제외한 2열의 성분들을 0으로 만든다 : 1행 - (1행 2열/2행 2열)*2행, 3행 – (3행 2열/2행 2열)*2행을 계산한 뒤 1,3행에 각각 대입한다

(1) $$X = \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 2 & 3 & -4 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 8 & -2 & 1 & 0 & 0 \\ 0 & 1\times 8 & 2\times 8 & -6\times 8 & 1\times 8 & 0 \\ 0 & 2 & 3 & -4 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & -18 & 49 & -8 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 2 & 3 & -4 & 0 & 1 \end{array}\right)$$

(2) $$X = \left(\begin{array}{ccc|ccc} -1 & 0 & 18 & 49 & -8 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 2 & 3 & -4 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & 18 & 49 & -8 & 0 \\ 0 & 1\times 2 & 2\times 2 & -6\times 2 & 1\times 2 & 0 \\ 0 & 2 & 3 & -4 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & -18 & 49 & -8 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right)$$

가우스 조던 소거법 공식

#4 3행 3열을 제외한 3열의 성분들을 0으로 만든다 : 1행 - (1행 3열/3행 3열)*3행, 2행 – (2행 3열/3행 3열)*3행을 계산한 뒤 1,2행에 각각 대입한다

$$(1)\ X = \left(\begin{array}{ccc|ccc} -1 & 0 & -18 & 49 & -8 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & -18 & 49 & -8 & 0 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 0 & -1\times 18 & 8\times 18 & -2\times 18 & 1\times 18 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right)$$

$$(2)\ X = \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 2 & -6 & 1 & 0 \\ 0 & 0 & -1\times -2 & 8\times -2 & -2\times -2 & 1\times -2 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right)$$

#5 우측 결과 행렬에 왼쪽 행렬의 대각 성분을 나누어 주어 최종 결과를 얻는다

$$(1)\ X = \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & -95\times -1 & 28\times -1 & -18\times -1 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 95 & -28 & 18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right)$$

$$(2)\ X = \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 0 & 10\times 1 & -3\times 1 & 2\times 1 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 95 & -28 & 18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right)$$

$$(3)\ X = \left(\begin{array}{ccc|ccc} -1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8 & -2 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & -95 & 28 & -18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & 8\times -1 & -2\times -1 & 1\times -1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 95 & -28 & 18 \\ 0 & 1 & 0 & 10 & -3 & 2 \\ 0 & 0 & -1 & -8 & 2 & -1 \end{array}\right)$$

가우스 조던 소거법 공식


```c
void adjust_3x6_mat(float (*A)[6], int idx, float (*R)[6])
{
    int i, j;
    float div_factor, scale;

        scale = A[idx][idx];

    for(i = idx + 1; i < 3; i++)
    {
        //div_factor = -A[idx][idx] / A[idx + 1][idx];
        //div_factor = -A[idx + 1][idx] / A[idx][idx];
        //div_factor = -A[i][0] / A[idx][0];
        div_factor = -A[i][idx] / A[idx][idx];
        printf("div_factor = %f\n", div_factor);

                if(div_factor == 0.0)
                        continue;

        for(j = 0; j < 6; j++)
            R[i][j] = A[idx][j] * div_factor + A[i][j];
    }

        for(j = 0; j < 6; j++)
            R[idx][j] = A[idx][j] / scale;
}
```

→ idx의 열 정보를 가지고 idx행 idx열을 제외한 나머지 행, 열의 성분을 0으로 만든다

→ 우측 결과 행렬에 왼쪽 행렬의 대각 성분을 나누어 주어
단위행렬 | 역행렬의 형태로 만든다

OpenGL을 이용한 Gram-Schmidt 정규직교화

- gluLookAt, glColor3f, glVertex3f 함수

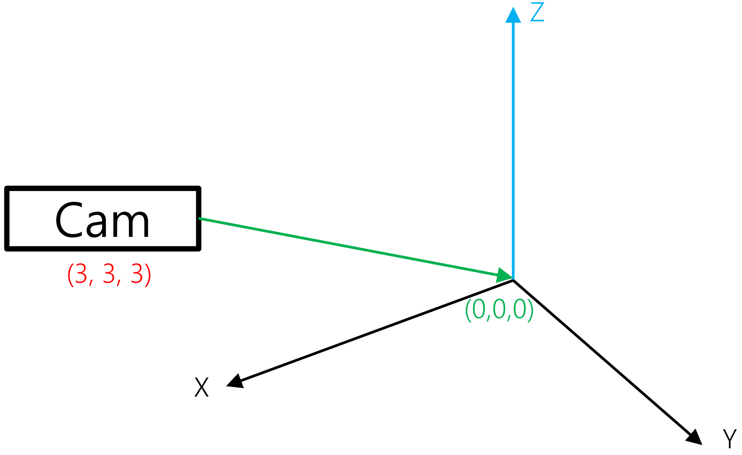
(1) gluLookAt 함수 : gluLookAt(카메라 x위치, 카메라 y위치, 카메라 z위치, 시점 x위치, 시점 y위치, 시점 z위치, 수직 x축, 수직 y축, 수직 z축)

카메라(사용자의 눈?) 위치 : glwindow의 3차원 좌표에서 카메라의 위치를 의미한다

시점 : 카메라가 좌표상 바라보는 좌표점

수직축 : 카메라와 평행한 축을 의미한다

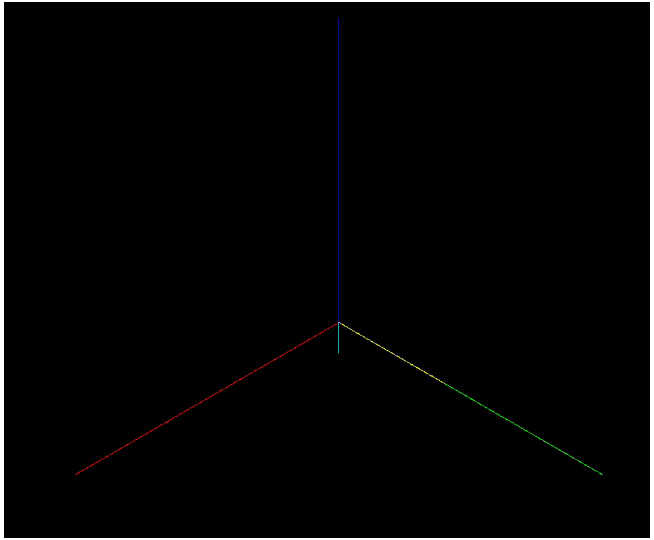
*예 :



→ 옆 그림과 같이 설정한다면
→ gluLookAt(3, 3, 3, 0, 0, 0, 0, 0, 1)

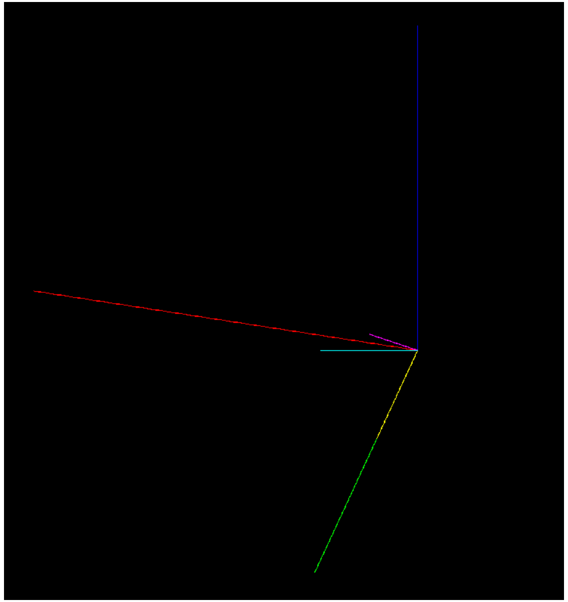
OpenGL을 이용한 Gram-Schmidt 정규직교화

(2) gluLookAt 함수 설정에 따른 opengl 화면



```
#if 0
        gluLookAt(3.0, 3.0, 3.0, 0, 0, 0, 0, 0, 1);
#endif
#if 0
```

* (3,3,3)위치에서 z축 중심으로 60도 회전



```
#if 1
        //z-axis 60deg
        gluLookAt(-1.098, 4.098, 3, 0, 0, 0, 0, 0, 1);
#endif
```

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos 60 & -\sin 60 & 0 \\ \sin 60 & \cos 60 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}$$

OpenGL을 이용한 Gram-Schmidt 정규직교화

(3) glColor3f, glVertex3f, glTranslate함수

→ glColor3f(Red, Green, Blue) : 그리고자 하는 vertex의 색깔을 설정한다

→ glVertex3f(x, y, z) : 좌표 평면상의 설정된 위치에 점을 그린다

* 직선을 그리기 위해선 시작점과 끝점을 설정한다

끝점설정

시작점 설정

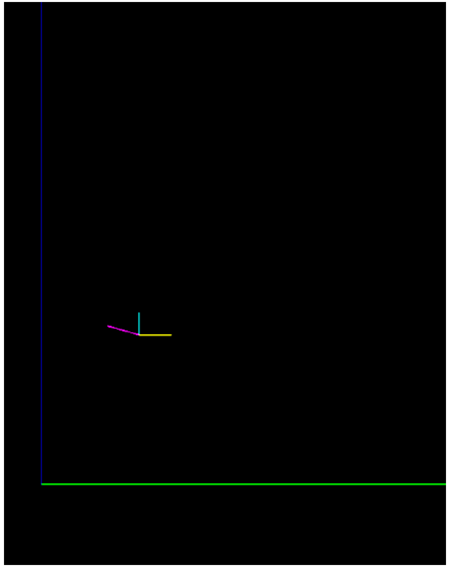
```
    //x-axis
glColor3f (1.0, 0.0, 0.0);
glVertex3f(100.0, 0.0, 0.0);
glVertex3f(0.0, 0.0, 0.0);

    //y-axis
glColor3f (0.0, 1.0, 0.0);
glVertex3f(0.0, 100.0, 0.0);
glVertex3f(0.0, 0.0, 0.0);

    //z-axis
glColor3f (0.0, 0.0, 1.0);
glVertex3f(0.0, 0.0, 100.0);
glVertex3f(0.0, 0.0, 0.0);
glEnd();
```

→ glTranslate : object의 평행이동

```
//vector를 offset만큼 수평 이동시킨다
glTranslatef(offset.x, offset.y, offset.z);
```



OpenGL을 이용한 Gram-Schmidt 정규직교화

- 정규직교화 결과
  - → V1 = (0, 40, 0); V2 = (20, 20, 10); V3 = (10, 10, 10);   → X : red, Y : Green, Z : Blue