

1) Hãy cho biết ma trận nào dưới đây là ma trận phép biến đổi đối xứng qua đường thẳng  $y = -x$  trong không gian 2 chiều:

A.  $\begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$  B.  $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$  C.  $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

D. Cả A, B và C đều sai

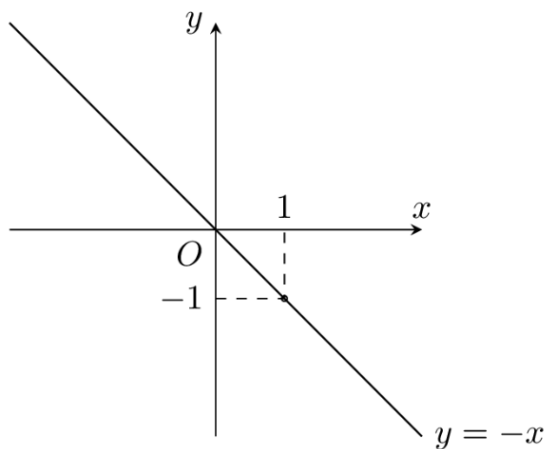
- Xoay 45 độ ngược chiều kim đồng hồ
- Đối xứng qua trục x
- Xoay 45 độ cùng chiều kim đồng hồ

$$\begin{pmatrix} 2^{0.5/2} & 2^{0.5/2} & 0 \\ -2^{0.5/2} & 2^{0.5/2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 2^{0.5/2} & -2^{0.5/2} & 0 \\ 2^{0.5/2} & 2^{0.5/2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$: \begin{pmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



1) Hãy cho biết ma trận nào dưới đây là ma trận phép biến đổi đối xứng qua đường thẳng  $y = -x$  trong không gian 2 chiều:

A.  $\begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$  B.  $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$  C.  $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

D. Cả A, B và C đều sai

Thử với điểm A(1, -2) thì kết quả phải là (2, -1).

Nhân với cả 3 ma trận A, B, C đều không cho kết quả (2, -1)

**2) Trong mô hình cộng màu (additive color), người ta tạo ra các màu sắc khác nhau từ 3 màu cơ bản nào dưới đây:**

- A.** Màu đỏ(red), màu lục (green), màu xanh (blue)
- B. Màu xanh lơ (cyan), màu đỏ thẫm (magenta), màu vàng (yellow)
- C. Màu đỏ(red), màu xanh lơ (cyan), màu xanh (blue)
- D. Màu đỏ(red), màu lục (green), màu xanh (vàng)

**3) Hàm nào sau đây thiết lập thể tích nhìn cho phép chiếu xiên (oblique projection)**

- A.glOrtho      B.glFrustum      C.gluPerspective      **D.Cả A, B, C đều sai**

**4) Cắt xén đoạn thẳng AB với thể tích nhìn chuẩn (CVV), trong trường hợp xấu nhất, cần phải tìm giao điểm của AB với các mặt của CVV bao nhiêu lần**

- A.2 lần      B. 4 lần      C. 5 lần      **D. 6 lần**

**5) Trong số những bộ đệm dưới đây**

- (i) Bộ đệm màu sắc (color buffer)
- (ii) Bộ đệm chiều sâu (depth buffer)
- (iii) Bộ đệm tích lũy (accumulation buffer)

OpenGL sử dụng loại bộ đệm nào

- A. Chỉ sử dụng (i)
- B. Chỉ sử dụng (i) và (iii)
- C. Chỉ sử dụng (i) và (ii)
- D. Sử dụng cả 3 loại bộ đệm trên

**6) Ta có hàm mymouse được khai báo như sau:**

```
void mymouse(int btn, int state, int x, int y)
```

Trong hàm main() gọi hàm đăng ký sự kiện bấm chuột như sau:

```
glutMouseFunc(mymouse).
```

Hãy cho biết khi chạy chương trình, mỗi lần bấm chuột hàm mymouse sẽ được gọi bao nhiêu lần.

- A. 1 lần
- B. 2 lần
- C. 3 lần
- D. 4 lần

**10) Để thiết lập thể tích nhìn cho phép chiếu trực giao ta gọi hàm glMatrixMode() với tham số là**

- A. GL\_PROJECTION
- B. GL\_MODELVIEW
- C. GL\_ORTHO
- D. GL\_PERSPECTIVE

- 11) Để thực hiện phép biến đổi đối xứng qua gốc tọa độ, ta gọi hàm nào sau đây  
A.glRotatef() B.glScalef() C.Cả A, B đều đúng D.Cả A, B đều sai
- 12) Phép chiếu trục đo (axonometric) thuộc dạng phép chiếu nào  
A.Phép chiếu xiên B.Phép chiếu trục giao C.Phép chiếu phối cảnh D.Cả A, B, C đều sai
- 13) Trong dạng biểu diễn đồng nhất của một điểm trong không gian 3 chiều, thì  
A.Thành phần tọa độ thứ 4 có giá trị 1 B. Thành phần tọa độ thứ 4 có giá trị 0  
C.Thành phần tọa độ thứ 4 có giá trị dương D. Thành phần tọa độ thứ 4 có giá trị âm
- 14) Nhận định nào dưới đây về thể tích nhìn chuẩn (CVV) là đúng  
A.Là hình lập phương có cạnh dài 1 đơn vị. B.Là hình lập phương có cạnh dài 2 đơn vị  
C.Có đỉnh trùng với gốc tọa độ D. Có 2 trong số 3 đáp án trên là đúng
- 15) Thành phần ánh sáng nào dưới đây phụ thuộc vào vị trí camera  
A.Ánh sáng môi trường B.Ánh sáng khuếch tán  
C.Ánh sáng phản chiếu D.Cả A và B đều đúng

**1) Giả sử màn hình có độ phân giải là  $1024 \times 768$ , biết rằng bộ đệm frame có  $b = 12$  mặt phẳng bit, màn hình sử dụng LUT có độ rộng  $w = 20$ .**

Dung lượng của bộ đệm frame là \_\_\_\_\_ bytes

Dung lượng của LUT là \_\_\_\_\_ bytes

Màn hình có thể hiển thị được \_\_\_\_\_ màu cùng một lúc

- Dung lượng frame buffer =  $1024 \times 768 \times 12 / 8 = 1\,179\,648$  bytes
- Dung lượng LUT =  $2^{12} \times 20 / 8 = 10\,240$  bytes
- Màn hình có thể hiển thị được  $2^{12} = 4096$  màu cùng lúc

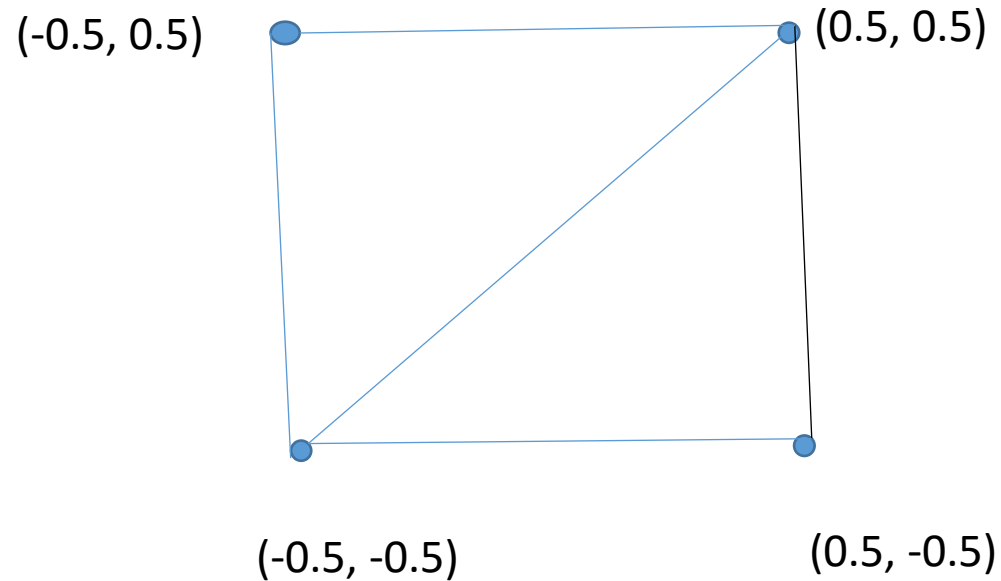
**2) Biết rằng cửa sổ màn có kích thước là  $600 \times 800$ . Giả sử ta muốn vẽ hình ảnh vào trong khung nhìn là hình vuông có kích thước 300 nằm giữa cửa sổ màn hình. Hãy thiết lập tham số cho hàm `glViewport(_____)`**

- Chiều rộng  $w = 300$
- Chiều cao  $h = 300$
- $X = (600 - 300)/2 = 150$
- $Y = (800 - 300)/2 = 250$
- `glViewport(150, 250, 300, 300)`



### 3) Đoạn mã lệnh sau vẽ bao nhiêu tam giác:

```
glPolygonMode(GL_FRONT_AND_BACK, GL_LINE);  
glBegin(GL_TRIANGLE_STRIP);  
    glVertex2f(0.5f, -0.5f); glVertex2f(0.5f, 0.5f);    glVertex2f(-0.5f, -0.5f);  
    glVertex2f(-0.5f, 0.5f); glVertex2f(0.5f, 0.5f);  
glEnd();
```





4) Hãy viết prototype của hàm xử lý sự kiện bấm phím trong GLUT

---

`void myKeyboard(unsigned char key, int x, int y);`

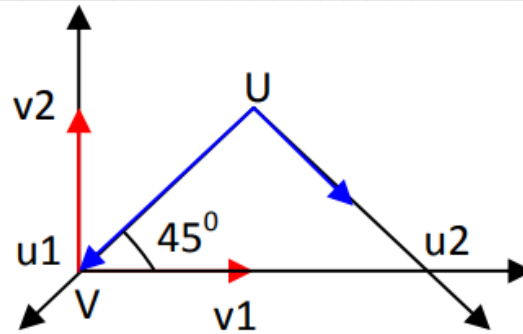
6) Cho hai điểm  $A = (1, 2, 3)$ ,  $B = (5, 6, 7)$ . Hãy viết phương trình tham số của tia đi qua A và B có điểm gốc tại A

---

$$L(t) = A + (B-A)t = (1, 2, 3) + (4, 4, 4)t, \text{ với } t \geq 0$$

$$L(t) = (1 + 4t, 2 + 4t, 3 + 4t), \text{ với } t \geq 0$$

7) Cho hệ trục tọa độ hai chiều thứ nhất ( $Vv_1v_2$ ) với các vector cơ sở là  $v_1 = [1, 0]$ ,  $v_2 = [0, 1]$ . Cho một hệ tọa độ hai chiều thứ hai ( $Uu_1u_2$ ) với các vector cơ sở  $u_1, u_2$  như trong hình vẽ, gốc tọa độ U ở vị trí (1, 1). Lưu ý:  $u_1$  có chiều dài gấp đôi  $u_2$ . Một điểm C có tọa độ (1, 2) trong hệ tọa độ thứ nhất. Hãy cho biết tọa độ của điểm C trong hệ tọa độ thứ hai \_\_\_\_\_



- $u_1 = -v_1 - v_2$
- $u_2 = v_1/2 - v_2/2$
- $U = v_1 + v_2 + V$

- $b = (M^T)^{-1}a = (-0.5, -1, 1)$

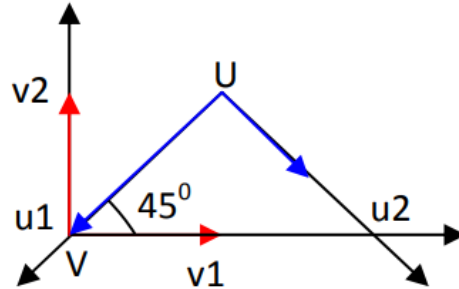
$$M = \begin{pmatrix} -1 & -1 & 0 \\ 0.5 & -0.5 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

$$M^T = \begin{pmatrix} -1 & 1/2 & 1 \\ -1 & -1/2 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$(M^T)^{-1} = \begin{pmatrix} \frac{-1}{2} & \frac{-1}{2} & 1 \\ 1 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

7) Cho hệ trục tọa độ hai chiều thứ nhất ( $Vv_1v_2$ ) với các vector cơ sở là  $v_1 = [1, 0]$ ,  $v_2 = [0, 1]$ . Cho một hệ tọa độ hai chiều thứ hai ( $Uu_1u_2$ ) với các vector cơ sở  $u_1, u_2$  như trong hình vẽ, gốc tọa độ  $U$  ở vị trí (1, 1). Lưu ý:  $u_1$  có chiều dài gấp đôi  $u_2$ .

Một điểm  $C$  có tọa độ (1, 2) trong hệ tọa độ thứ nhất. Hãy cho biết tọa độ của điểm  $C$  trong hệ tọa độ thứ hai \_\_\_\_\_



$$u_1 = -v_1 - v_2 \quad \rightarrow v_1 = -u_1/2 + u_2$$

$$u_2 = v_1/2 - v_2/2 \quad \rightarrow v_2 = -u_1/2 - u_2$$

$$U = v_1 + v_2 + V \quad \rightarrow V = -v_1 - v_2 + U = u_1 + U$$

$$C = v_1 + 2v_2 + V = -u_1/2 + u_2 - u_1 - 2u_2 + u_1 + U = -(1/2)u_1 - u_2 + U$$

$$\rightarrow C(-0.5, -1, 1)$$

**8) Cho điểm A trong không gian hai chiều, biết tọa độ của A = (3, 4). Quay điểm A quanh điểm B = (1, 1) một góc 45 độ cùng chiều kim đồng hồ ta được điểm A'. Hãy cho biết tọa độ của A' =**

$$= \begin{pmatrix} \cos(\theta) & -\sin(\theta) & d_x \\ \sin(\theta) & \cos(\theta) & d_y \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{aligned} d_x &= -V_x \cos(\theta) + V_y \sin(\theta) + V_x \\ d_y &= -V_x \sin(\theta) - V_y \cos(\theta) + V_y \end{aligned}$$

$$\begin{pmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & -\sqrt{2}+1 \\ \frac{-\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \frac{5\sqrt{2}+2}{2} \\ \frac{\sqrt{2}+2}{2} \\ 1 \end{pmatrix}$$

9) Giả sử ta sử dụng `gluLookAt(1.0f, 0, 1.0f, 0, 0, 0, 0, 0, 1.0f)`; để thiết lập camera. Hãy tìm ma trận  $V$  chuyển đổi từ hệ tọa độ thế giới sang hệ tọa độ camera.

eye, look at, up  $\rightarrow$   $\mathbf{u}$ ,  $\mathbf{v}$ ,  $\mathbf{n}$

$\mathbf{n} = \text{eye} - \text{look}$ .

$\mathbf{u} = \mathbf{up} \times \mathbf{n}$ ,

$\mathbf{v} = \mathbf{n} \times \mathbf{u}$

$\mathbf{u}$ ,  $\mathbf{v}$ ,  $\mathbf{n}$  : unit vector

$$V = \begin{pmatrix} u_x & u_y & u_z & d_x \\ v_x & v_y & v_z & d_y \\ n_x & n_y & n_z & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(d_x, d_y, d_z) = (-\text{eye} \cdot \mathbf{u}, -\text{eye} \cdot \mathbf{v}, -\text{eye} \cdot \mathbf{n})$$

- $\mathbf{n} = (1, 0, 1) - (0, 0, 0) = (1, 0, 1)$
- $\mathbf{u} = (0, 0, 1) \times (1, 0, 1) = (0, 1, 0)$
- $\mathbf{v} = (1, 0, 1) \times (0, 1, 0) = (-1, 0, 1)$
- Chuẩn hóa  $\mathbf{u}$ ,  $\mathbf{v}$ ,  $\mathbf{n}$

```
0, 1, 0, 0,  
-0.707107, 0, 0.707107, 0,  
0.707107, 0, 0.707107, -1.41421,  
0, 0, 0, 1,
```

# 10) Giả sử ta dùng những câu lệnh sau để thiết lập thể tích nhìn

```
glMatrixMode(GL_PROJECTION);  
glLoadIdentity();  
glOrtho(-1.3, 1.7, -2.4, 2.6, 1, 101);
```

Hãy tìm ma trận chiếu (projection matrix) để biến đổi thể tích nhìn thành thể tích nhìn chuẩn

$$\begin{bmatrix} \frac{2}{right-left} & 0 & 0 & -\frac{right+left}{right-left} \\ 0 & \frac{2}{top-bottom} & 0 & -\frac{top+bottom}{top-bottom} \\ 0 & 0 & \frac{2}{near-far} & -\frac{far+near}{far-near} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
0.666667, 0, -0, -0.133333,  
0, 0.4, -0, -0.04,  
0, 0, -0.02, -1.02,  
0, 0, -0, 1,
```

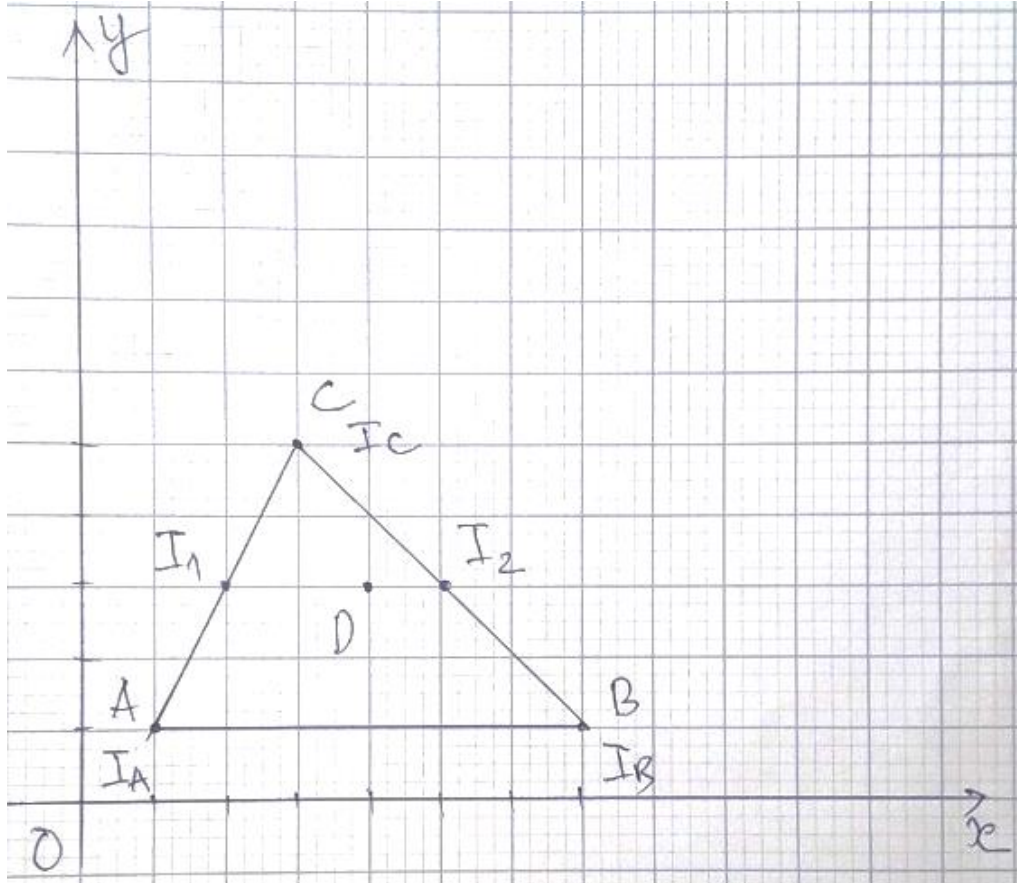
**11) Cho mặt  $\left(\frac{x}{6}\right)^2 + \left(\frac{y}{8}\right)^2 + \left(\frac{z}{10}\right)^2 = 3$  và một nguồn sáng đặt tại điểm có tọa độ  $S (9, 8, 14)$ . Hãy xác định thành phần ánh sáng khuếch tán tại điểm  $P (6, 8, 10)$  của mặt khi cho biết cường độ  $I_s$  của nguồn sáng và hệ số phản xạ khuếch tán  $\rho_d$  của bề mặt \_\_\_\_\_**

$$I_d = I_s \rho_d \frac{s \bullet m}{|s||m|}$$

- $n = (dF/dx, dF/dy, dF/dz) = (x/3, y/4, z/5)$
- $n(6, 8, 10) = (2, 2, 2)$
- $s = (9, 8, 14) - (6, 8, 10) = (3, 0, 4)$
- $I_d = I_s \rho_d (2 \times 3 + 2 \times 4) / (5 \times \sqrt{12}) = 0.81 I_s \rho_d$



**12) Giả sử sử dụng phương pháp nội suy Gouraud đối với tam giác ABC. Biết rằng tọa độ các đỉnh của tam giác như sau:  $A=(1, 1)$ ,  $B=(7, 1)$ ,  $C=(3, 5)$ , cường độ ánh sáng tại các đỉnh A, B, C lần lượt là  $I_A$ ,  $I_B$ ,  $I_C$ . Cường độ ánh sáng tại điểm  $D=(4, 3)$  là \_\_\_\_\_**



- $I_1 = (I_a + I_b)/2$
- $I_2 = (I_b + I_c)/2$
- $I_d = (2 \times I_2 + I_1)/3 = (I_a + 3I_b + 2I_c)/3$
- Tính cường độ ánh sáng tại  $E(11/3, 7/3)$

Cho đoạn thẳng AB trong không gian đối tượng (Object Space) như sau: A (1, 0, 0), B (-1, -1, 1). Các đỉnh của đoạn thẳng được biến đổi vào không gian thế giới (World Space) bằng cách thực hiện lần lượt các phép biến đổi sau:

- Quay quanh trục Y một góc 45 độ ngược chiều kim đồng hồ.
- Quay tiếp quanh trục Y một góc 45 độ ngược chiều kim đồng hồ.
- Tịnh tiến theo trục X và trục Y : T(1, 2, 0)

Sử dụng hàm `gluLookAt()` để thiết lập Camera với tham số như sau:

```
gluLookAt(-5, 0, 5, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);
```

Thiết lập phép chiếu phối cảnh (Perspective Viewing) bằng hàm `glFrustum()` với các tham số như sau:

```
glFrustum(-5, 5, -5, 5, 5, 15);
```

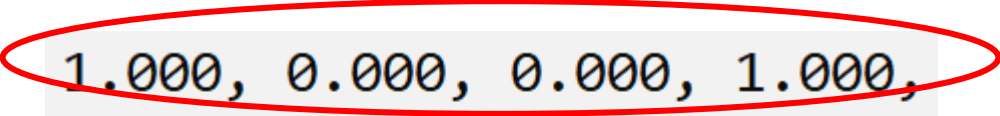
- 1) Tính ma trận Model để chuyển các đỉnh từ không gian đối tượng vào không gian thế giới
- 2) Tính ma trận View để chuyển các đỉnh từ không gian thế giới vào không gian Camera
- 3) Tính ma trận Model-View để chuyển từ không gian đối tượng vào không gian Camera.
- 4) Tính ma trận chiếu (Projection Matrix) để đưa các đỉnh vào không gian cắt
- 5) Tính tọa độ trong không gian thế giới, không gian Camera và không gian cắt của đỉnh B.

- Quay quanh trục Y một góc 45 độ ngược chiều kim đồng hồ.
- Quay tiếp quanh trục Y một góc 45 độ ngược chiều kim đồng hồ.
- Tịnh tiến theo trục X và trục Y : T(1, 2, 0)

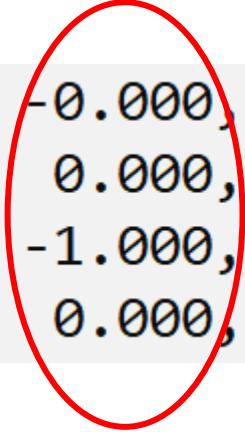
$$R_y(\beta) = \begin{pmatrix} c & 0 & s & 0 \\ 0 & 1 & 0 & 0 \\ -s & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 & m_{14} \\ 0 & 1 & 0 & m_{24} \\ 0 & 0 & 1 & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
-0.000, 0.000, 1.000, 0.000,
0.000, 1.000, 0.000, 0.000,
-1.000, 0.000, -0.000, 0.000,
0.000, 0.000, 0.000, 1.000,
```

```
1.000, 0.000, 0.000, 1.000,
0.000, 1.000, 0.000, 2.000,
0.000, 0.000, 1.000, 0.000,
0.000, 0.000, 0.000, 1.000,
```



1.000,	0.000,	0.000,	1.000,
0.000,	1.000,	0.000,	2.000,
0.000,	0.000,	1.000,	0.000,
0.000,	0.000,	0.000,	1.000,



-0.000,	0.000,	1.000,	0.000,
0.000,	1.000,	0.000,	0.000,
-1.000,	0.000,	-0.000,	0.000,
0.000,	0.000,	0.000,	1.000,

-0.000,	0.000,	1.000,	1.000,
0.000,	1.000,	0.000,	2.000,
-1.000,	0.000,	-0.000,	0.000,
0.000,	0.000,	0.000,	1.000,

Sử dụng hàm gluLookAt() để thiết lập Camera với tham số như sau:

```
gluLookAt(-5, 0, 5, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);
```

eye, look at, up  $\rightarrow$  u, v, n

**n** = eye – look.

**u** = **up**  $\times$  **n**,

**v** = **n**  $\times$  **u**

**u, v, n** : unit vector

$$V = \begin{pmatrix} u_x & u_y & u_z & d_x \\ v_x & v_y & v_z & d_y \\ n_x & n_y & n_z & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(d_x, d_y, d_z) = (-eye \bullet u, -eye \bullet v, -eye \bullet n)$$

View Matrix

```
0.707, 0.000, 0.707, 0.000,  
0.000, 1.000, 0.000, 0.000,  
-0.707, 0.000, 0.707, -7.071,  
0.000, 0.000, 0.000, 1.000,
```



```
-0.000, 0.000, 1.000, 1.000,  
0.000, 1.000, 0.000, 2.000,  
-1.000, 0.000, -0.000, 0.000,  
0.000, 0.000, 0.000, 1.000,
```

### View Matrix

```
0.707, 0.000, 0.707, 0.000,  
0.000, 1.000, 0.000, 0.000,  
-0.707, 0.000, 0.707, -7.071,  
0.000, 0.000, 0.000, 1.000,
```

### Model-View Matrix

```
-0.707, 0.000, 0.707, 0.707,  
0.000, 1.000, 0.000, 2.000,  
-0.707, 0.000, -0.707, -7.778,  
0.000, 0.000, 0.000, 1.000,
```

Thiết lập phép chiếu phối cảnh (Perspective Viewing) bằng hàm `glFrustum()` với các tham số như sau:

```
glFrustum(-5, 5, -5, 5, 5, 15);
```

$$\begin{bmatrix} \frac{2}{right-left} & 0 & 0 & -\frac{right+left}{right-left} \\ 0 & \frac{2}{top-bottom} & 0 & -\frac{top+bottom}{top-bottom} \\ 0 & 0 & \frac{2}{near-far} & -\frac{far+near}{far-near} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Projection Matrix

```
1.000, 0.000, 0.000, 0.000,
0.000, 1.000, 0.000, 0.000,
0.000, 0.000, -2.000, -15.000,
0.000, 0.000, -1.000, 0.000,
```



```
AWorld =    1.000, 2.000, -1.000, 1.000,  
AView =   -0.000, 2.000, -8.485, 1.000,  
AClip =   -0.000, 2.000, 1.971, 8.485,  
AClip =   -0.000, 0.236, 0.232, 1.000,  
BWorld =    2.000, 1.000, 1.000, 1.000,  
BView =    2.121, 1.000, -7.778, 1.000,  
BClip =    2.121, 1.000, 0.556, 7.778,  
BClip =    0.273, 0.129, 0.072, 1.000,
```