****

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

****-----🙞🙜🕮🙞🙜-----

**BÀI TẬP THỰC NGHIỆM**

**ĐỒ HOẠ MÁY TÍNH**

**Đề tài: Mô phỏng một phòng học lý thuyết thuộc nhà A9 trường ĐH Công Nghiệp Hà Nội**

Giảng viên hướng dẫn: Ths. Vũ Minh Yến

Nhóm thực hiện: Nhóm 4

Lớp: 20201IT6010003 Khóa: K15

Thành viên trong nhóm :

Nguyễn Duy Hùng – **2020600796**

Lương Nguyễn Anh Đào – **2020601077**

Nguyễn Văn Minh - **2020600499**

**Hà nội, năm 2022**

Mục Lục

[**LỜI NÓI ĐẦU 3**](#_Toc123079200)

[**Chương 1: Xác định và phân tích bài toán 4**](#_Toc123079201)

[**1.1. Bài toán 4**](#_Toc123079202)

[**1.2. Mô tả đối tượng cần thiết kế 4**](#_Toc123079203)

[**1.2.1. Cửa ra vào 4**](#_Toc123079204)

[**1.2.2. Cửa sổ 5**](#_Toc123079205)

[**1.2.3. Quạt 5**](#_Toc123079206)

[**1.2.4. Điều hòa 5**](#_Toc123079207)

[**1.2.5. Bàn giáo viên 6**](#_Toc123079208)

[**1.2.6. Ghế giáo viên 7**](#_Toc123079209)

[**1.2.7. Bục giảng 7**](#_Toc123079210)

[**1.2.8. Bảng 7**](#_Toc123079211)

[**1.2.9. Bàn học sinh 8**](#_Toc123079212)

[**1.3. Mô tả bố cục khung cảnh chung 9**](#_Toc123079213)

[**1.4. Mô tả kịch bản của chương trình 9**](#_Toc123079214)

[**a, Hoạt động phổ biến 9**](#_Toc123079215)

[**b, Mô tả chi tiết từng hoạt động 10**](#_Toc123079216)

[**c, Các phép biến đổi được sử dụng 11**](#_Toc123079217)

[**Chương 2 : Cài đặt chương trình 12**](#_Toc123079218)

[**2.1. Kỹ thuật tạo mô hình Cửa ra vào 12**](#_Toc123079219)

[**2.2. Kỹ thuật tạo mô hình Cửa sổ 13**](#_Toc123079220)

[**2.3. Kỹ thuật tạo mô hình Điều hòa 14**](#_Toc123079221)

[**2.4. Kỹ thuật tạo mô hình Quạt 15**](#_Toc123079222)

[**2.5. Kỹ thuật tạo mô hình Ghế giáo viên 17**](#_Toc123079223)

[**2.6. Kỹ thuật tạo mô hình Ngăn kéo 19**](#_Toc123079224)

[**2.7. Kỹ thuật tạo mô hình Tủ bàn giáo viên 20**](#_Toc123079225)

[**2.8. Kỹ thuật điều chỉnh camera 21**](#_Toc123079226)

[**2.9. Kỹ thuật chiếu sáng 21**](#_Toc123079227)

[**2.10. Kỹ thuật tạo mô hình Bàn giáo viên 22**](#_Toc123079228)

[**2.11. Kỹ thuật tạo mô hình Bàn học sinh 22**](#_Toc123079229)

[**2.12. Kỹ thuật tạo mô hình Bục giảng 24**](#_Toc123079230)

[**2.13. Kỹ thuật tạo mô hình Bảng 24**](#_Toc123079231)

[**Chương 3 : Kết quả đạt được 26**](#_Toc123079232)

[**3.1. Mô hình Điều hòa 26**](#_Toc123079233)

[**3.2. Mô hình Quạt 26**](#_Toc123079234)

[**3.3. Mô hình Bàn giáo viên có đủ ngăn kéo, tủ bàn 27**](#_Toc123079235)

[**3.4. Mô hình Ghế giáo viên 28**](#_Toc123079236)

[**3.5. Mô hình Cửa sổ 29**](#_Toc123079237)

[**3.6. Mô hình Cửa ra vào 30**](#_Toc123079238)

[**3.7. Mô hình Bàn học sinh 31**](#_Toc123079239)

[**3.8. Mô hình Bảng 32**](#_Toc123079240)

[**3.9. Điều khiển camera 33**](#_Toc123079241)

[**3.10. Mô hình mô phỏng toàn bộ phòng học lý thuyết thuộc nhà A9 33**](#_Toc123079242)

[**KẾT LUẬN 36**](#_Toc123079243)

[**TÀI LIỆU THAM KHẢO 37**](#_Toc123079244)

# LỜI NÓI ĐẦU

Đồ họa máy tính là một lĩnh vực của khoa học máy tính nghiên cứu về cơ sở toán học, các thuật toán cũng như các kĩ thuật để cho phép tạo, hiển thị và điều khiển hình ảnh trên màn hình máy tính.

Đồ họa máy tính ngày nay được ứng dụng rất rộng rãi trong nhiều lĩnh vực khoa học, kĩ thuật, nghệ thuật, kinh doanh, quản lí, … Các ứng dụng đồ họa rất đa dạng, phong phú và phát triển liên tục không ngừng. Ngày nay, hầu như không có chương trình ứng dụng nào mà không sử dụng kĩ thuật đồ họa để làm tăng tính hấp dẫn của mình.

Việc mô tả dữ liệu thông qua các hình ảnh và màu sắc đa dạng của nó, các chương trình đồ họa thường thu hút người sử dụng bởi tính thân thiện, dễ dùng, … kích thích khả năng sáng tạo và nâng cao năng suất làm việc

Chúng em được tìm hiểu và làm quen với một số kĩ thuật đồ họa trên máy tính thông qua tài liệu, bài giảng do cô giáo Vũ Minh Yến giảng dạy. Với những kiến thức đã học chúng em quyết định chọn đề tài: **Mô phỏng một phòng học lý thuyết thuộc nhà A9 trường ĐH Công Nghiệp Hà Nội**

# Chương 1: Xác định và phân tích bài toán

## 1.1. Bài toán

Với sinh viên, có lẽ lớp học là một nơi rất thân thuộc. Đó là nơi các thầy, cô giáo mang đến cho chúng em những bài học mới, kiến thức mới, giảng giải kỹ càng cho chúng em. Nơi đó gắn liền với bàn học, bảng, bục giảng, quạt,…

Chính vì vậy, chúng em đã lên ý tưởng mô phỏng lại một phòng học lý thuyết thuộc nhà A9 trường Đại học Công nghiệp Hà Nội.

## 1.2. Mô tả đối tượng cần thiết kế

### 1.2.1. Cửa ra vào

* Cửa ra vào: Có kích thước với chiều dài là 2m7, chiều rộng là 60cm và chiều dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Núm cửa : Có kích thước với chiều dài 5cm, chiều rộng là 5mc và độ dày là 10cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Khung:
    - Trên: Có kích thước với chiều dài là 60cm, chiều rộng là 5cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
    - Dưới: Có kích thước với chiều dài là 60cm, chiều rộng là 5cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
    - Trái : Có kích thước với chiều dài là 1m4cm, chiều rộng là 5cm và độ dày là 5cm. Hệ trục tọa độ đặt tạị ¼ căn phòng.
    - Phải : Có kích thước với chiều dài là 1m4cm, chiều rộng là 5cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.2. Cửa sổ

* + Cánh trái: Có kích thước với chiều dài là 1m3, chiều rộng là 35cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Cánh phải: Có kích thước với chiều dài là 1m3, chiều rộng là 35cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.3. Quạt

* + Đế quạt: Có kích thước với chiều dài là 5cm, chiều rộng là 5cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Chân dọc của quat: Có kích thước với chiều dài là 20cm, chiều rộng là 5cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Cánh quạt: Có kích thước với chiều dài là 50cm, chiểu rộng là 20cm và độ dày là 1cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.4. Điều hòa

* Khung điều hòa:
  + Mặt dưới: Có kích thước với chiều dài là 1m, chiều rộng là 40cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt sau: Có kích thước với chiều dài là 1m, chiều rộng là 40cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt trên: Có kích thước với chiều dài là 1m, chiều rộng là 40cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt vuông phải: Có kích thước với chiều dài là 40cm, chiều rộng là 40cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt vuông trái: Có kích thước với chiều dài là 40cm, chiều rộng là 40cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt trước trên: Có kích thước với chiều dài là 1m, chiều rộng là 20cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Cánh đuôi: Có kích thước với chiều dài là 1m, chiều rộng là 20cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.5. Bàn giáo viên

* Khung bàn:
  + Mặt bàn: Có kích thước với chiều dài là 90cm, chiều rộng là 55cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt sau: Có kích thước với chiều dài là 90cm, chiều rộng là 50cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Chân bàn: Có kích thước với chiều dài là 50cm, chiều rộng là 40cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Tủ bàn:
  + Mặt tủ bên: Có kích thước với chiều dài là 40cm, chiều rộng là 30cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt tủ dưới: Có kích thước với chiều dài là 50cm, chiều rộng là 40cm và độ dày là 5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Cửa tủ: Có kích thước với chiều dài là 30cm, chiều rộng là 30cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Núm cửa tủ: Có kích thước với chiều dài là 7cm, chiều rộng là 3cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Ngăn kéo(Hộp bàn):
  + Mặt trước: Có kích thước với chiều dài là 30cm, chiều rộng là 15cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt sau: Có kích thước với chiều dài là 30cm, chiều rộng là 10cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt bên: Có kích thước với chiều dài là 30cm, chiều rộng là 10cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt bên: Có kích thước với chiều dài là 30cm, chiều rộng là 10cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Mặt đáy: Có kích thước với chiều dài là 30cm, chiều rộng là 30cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Núm tủ: Có kích thước với chiều dài là 5cm, chiều rộng là 2cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.6. Ghế giáo viên

* Phía dưới của ghế:
  + Đế: Có kích thước với chiều dài là 40cm, chiều rộng là 40cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Thanh dọc: Có kích thước với chiều dài là 30cm, chiều rộng là 6cm và độ dày là 6cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Phía trên của ghế:
  + Chỗ ngồi: Có kích thước với chiều dài là 50cm, chiều rộng là 50cm và độ dày là 4cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Tay ghế: Có kích thước với chiều dài là 50cm, chiều rộng là 20cm và độ dày là 6cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Lưng ghế: Có kích thước với chiều dài là 60cm, chiều rộng là 50cm và độ dày là 3cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.7. Bục giảng

* Bục: Có kích thước với chiều dài là 2m7, chiều rộng là 1m và độ dày là 20cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.8. Bảng

* Thân bảng: Có kích thước với chiều dài là 1m25, chiều rộng là 50m và độ dày là 1cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Khung chiều dài: Có kích thước với chiều dài là 1m25, chiều rộng là 2m và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Khung chiều rộng: Có kích thước với chiều dài là 50cm, chiều rộng là 2m và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

### 1.2.9. Bàn học sinh

* Bàn:
  + Mặt bàn: Có kích thước với chiều dài là 60cm, chiều rộng là 20cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Ngăn bàn: Có kích thước với chiều dài là 56cm, chiều rộng là 16cm và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Chân bàn: Có kích thước với chiều dài là 40cm, chiều rộng là 2cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Thanh ngang chân bàn: Có kích thước với chiều dài là 16cm, chiều rộng là 1,5cm và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Thanh nối 2 thang ngang: Có kích thước với chiều dài là 57cm, chiều rộng là 1,5cm và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Thanh nối bàn với ghế:
  + Thanh nối: Có kích thước với chiều dài là 20cm, chiều rộng là 1,5cm và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
* Ghế:
  + Mặt ghế: Có kích thước với chiều dài là 60cm, chiều rộng là 20cm và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Chân ghế: Có kích thước với chiều dài là 20cm, chiều rộng là 2cm và độ dày là 2cm. Hệ trục tọa độ đặt tại ¼ căn phòng.
  + Thanh ngang của chân ghế: Có kích thước với chiều dài là 16cm, chiều rộng là 1,5cm và độ dày là 1,5cm. Hệ trục tọa độ đặt tại ¼ căn phòng.

## 1.3. Mô tả bố cục khung cảnh chung

- Sàn của phòng: Có kích thước với chiều rộng là 3m05, chiều dài là 6m5 và độ sau là 5cm. Hệ trục tọa độ được đặt tại 1/4 căn phòng

- Hai mặt đối diện của phòng : Có kích thước với chiều rộng là 3m05, chiều dài là 5cm và chiều cao là 1m6. Hệ trục tọa độ được đặt tại 1/4 căn phòng

- Mặt trái của phòng: Có kích thước với chiều dài là 6m5, chiều cao là 1m6 và chiều dày là 5cm. Hệ trục tọa độ được đặt tại 1/4 căn phòng

- Mặt phải của phòng: Có kích thước với chiều dài là 6m5, chiều cao là 1m6 và chiều dày là 5cm. Hệ trục tọa độ được đặt tại 1/4 căn phòng

- Song cửa sổ: Có kích thước với chiều dài là 5cm, chiều rộng là 3cm và độ dày là 2cm. Hệ trục tọa độ được đặt tại 1/4 căn phòng.

## 1.4. Mô tả kịch bản của chương trình

### a, Hoạt động phổ biến

Nhóm chúng em tiến hành mô tả một số hoạt động của căn phòng học lý thuyết

* Phòng có thể xoay 3600 theo 3 trục x,y,z
* Quạt có thể quay 3600
* Điều hòa có thể mở ra mở vào
* Cửa số và cửa ra vào có thể mở ra và đóng lại
* Ngăn kéo có thể mở ra và đóng lại
* Cửa tủ bàn giáo viên có thể mở ra và đóng lại
* Ghế giáo viên có thể kéo ra, kéo vào, sang trái, sang phải và xoay

### b, Mô tả chi tiết từng hoạt động

Nhóm chúng em sử dụng keyboard để điều khiển từng chuyện động của các thành phần trong phòng

* Toàn bộ phòng:
  + Trục x : x,X
  + Trục y : y,Y
  + Trục z : z,Z
* Cửa ra vào :
  + Mở ra : c
  + Đóng vào : C
* Cửa sổ :
  + Cánh trái :
    - Mở ra : h
    - Đóng vào : H
  + Cánh phải :
    - Mở ra : j
    - Đóng vào : J
* Điều hòa :
  + Mở ra : f
  + Đóng vào : F
* Quạt :
  + Mở ra : m
  + Đóng vào : M
* Ghế giáo viên :
  + Sang trái : a
  + Sang phải : d
  + Kéo ghế ra : s
  + Kéo ghế vào : w
  + Xoay ghế : e,E
* Ngăn kéo :
  + Kéo ra : r
  + Kéo vào : t
* Cửa tủ bàn giáo viên :
  + Mở ra : o
  + Đóng vào : i

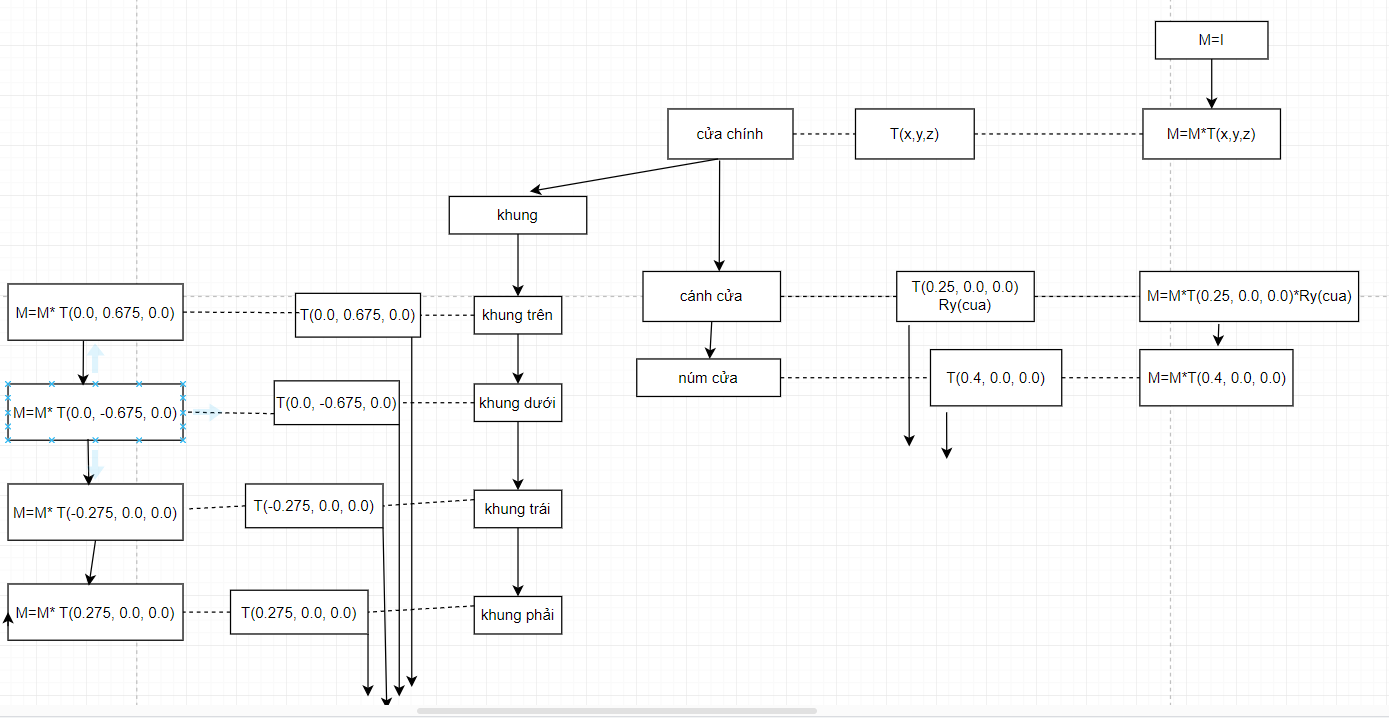
### c, Các phép biến đổi được sử dụng

* Phép quay
* Phép chiếu sáng
* Góc nhìn
* Vẽ hình lập phương đơn vị
* Vẽ các tam giác

# Chương 2 : Cài đặt chương trình

## 2.1. Kỹ thuật tạo mô hình Cửa ra vào

a. Mô hình phân cấp



b. Các mô tả vắn tắt cho việc tạo mô hình Cửa ra vào

//Cửa chính

//Khung cua

void Khung()

{

MakeColor(1, 0.8, 0.0, 1);

mat4 m = Translate(0.0, 0.675, 0.0) \* Scale(0.6, 0.05, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Cua\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(0.0, -0.675, 0.0) \* Scale(0.6, 0.05, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Cua\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(-0.275, 0.0, 0.0) \* Scale(0.05, 1.4, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Cua\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(0.275, 0.0, 0.0) \* Scale(0.05, 1.4, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Cua\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

//Cua

void Cua(GLfloat x, GLfloat y, GLfloat z)

{

Cua\_view = Translate(x,y,z) \* RotateY(90) \* Scale(0.6, 0.7, 0.85);

mat4 CanhCua\_view = Translate(0.25, 0.0, 0.0) \* RotateY(cua);

MakeColor(0.7, 0.8, 0.2, 1);

mat4 m = Translate(0.25, 0.0, 0.0) \* Scale(0.5, 1.3, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Cua\_view \* CanhCua\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Num cua

MakeColor(0.5, 0, 0.5, 1);

m = Translate(0.4, 0.0, 0.0) \* Scale(0.05, 0.05, 0.1);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Cua\_view \* CanhCua\_view \* m);

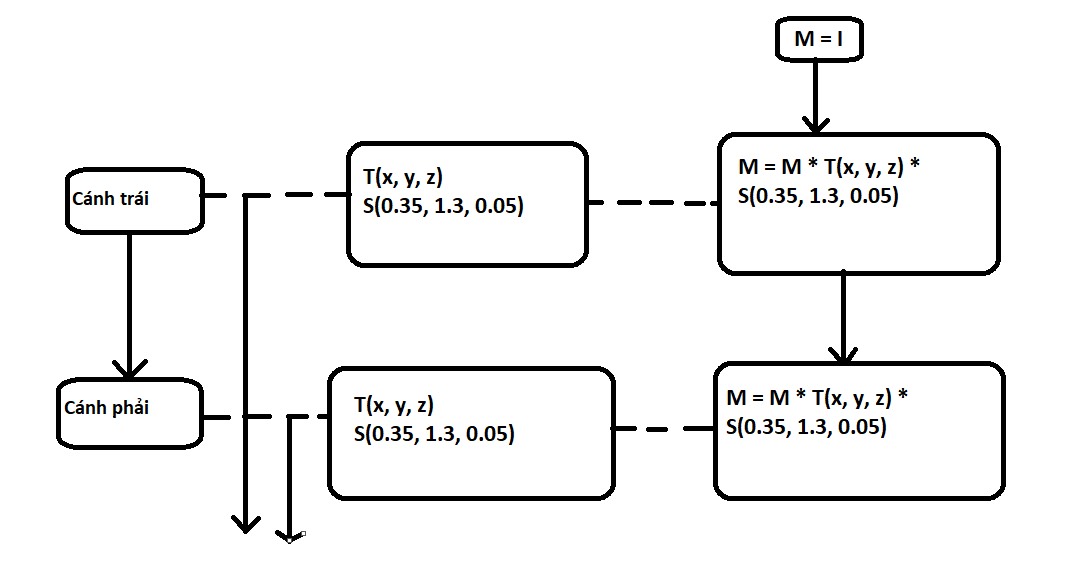
glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

Khung();

}

## 2.2. Kỹ thuật tạo mô hình Cửa sổ

a. Mô hình phân cấp



b. Các mô tả vắn tắt cho việc tạo mô hình Cửa sổ

GLfloat trai = 0.0, phai = 0.0;

// Cửa sổ

mat4 CuaSo\_view, CanhTrai\_view, CanhPhai\_view;

void CanhTrai(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.35, 1.3, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* CuaSo\_view \* CanhTrai\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void CanhPhai(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.35, 1.3, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* CuaSo\_view \* CanhPhai\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void CuaSo(GLfloat x, GLfloat y, GLfloat z)

{

MakeColor(0.5, 0.3, 0.3, 1);

CuaSo\_view = Translate(x,y,z) \* Scale(0.5, 0.4, 0.6) \* RotateY(90);

CanhTrai\_view = Translate(-0.35, 0.0, 0.0) \* RotateY(trai);

CanhPhai\_view = Translate(0.35, 0.0, 0.0) \* RotateY(phai);

CanhTrai(0.175, 0.0, 0.0);

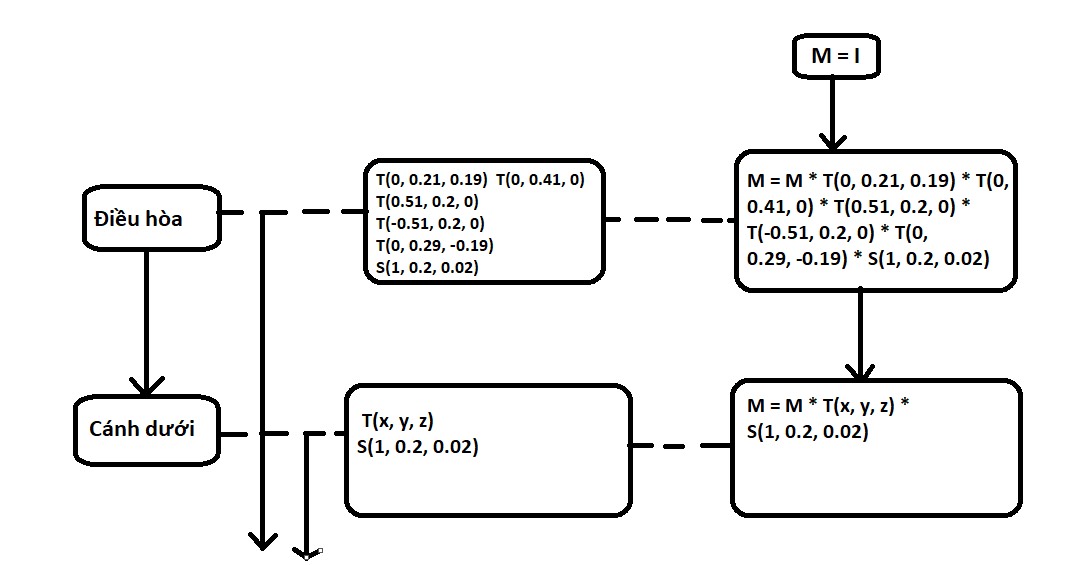
CanhPhai(-0.175, 0.0, 0.0);

//-1.51, 0.15, -0.805

}

## 2.3. Kỹ thuật tạo mô hình Điều hòa

a. Mô hình phân cấp



b. Các vắn tắt mô tả cho việc tạo mô hình Điều hòa

//Điều hoà

mat4 dieuHoa\_view;

mat4 model\_view\_duoi;

void dieuHoa()

{

mat4 m = Scale(1, 0.02, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* m);// duoi

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(0, (0.4 / 2 + 0.01), (0.4 / 2 - 0.01)) \* Scale(1, 0.4, 0.02);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* m);// sau

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(0, (0.4 + 0.01), 0) \* Scale(1, 0.02, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* m);// tren

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate((0.5 + 0.01), (0.4 / 2), 0) \* Scale(0.02, 0.4, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* m);// mat vuong phai

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(-(0.5 + 0.01), (0.4 / 2), 0) \* Scale(0.02, 0.4, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* m);// mat vuong trai

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

m = Translate(0, 0.3 - 0.01, -(0.2 - 0.01)) \* Scale(1, 0.2, 0.02); //0, (0.2/2 + 0.01), (0.2 - 0.01)

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* m);// mặt trước trên

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void canhduoi(GLfloat x, GLfloat y, GLfloat z)

{

MakeColor(0.0, 0.8, 0.8, 1);

mat4 m = Translate(x, y, z) \* Scale(1, 0.2, 0.02);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* dieuHoa\_view \* model\_view\_duoi \* m);// mặt trước dưới

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

GLfloat rdieuhoa = 0;

void dieuHoaFull(GLfloat x, GLfloat y, GLfloat z)

{

dieuHoa\_view = Translate(x,y,z) \* RotateY(-90) \* Scale(0.5, 0.4, 0.4);

dieuHoa();

model\_view\_duoi = Translate(0, 0, -0.2) \* RotateX(rdieuhoa) \* Translate(0, 0, 0.2);

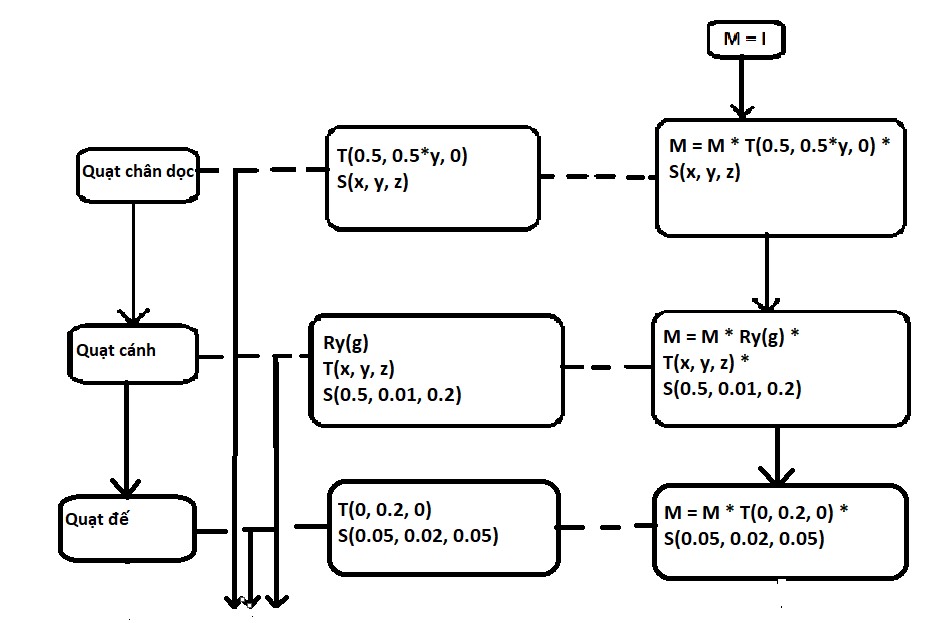
canhduoi(0, 0.101, -(0.2 - 0.01));

//-1.35, 0.6, 0-0.7

}

## 2.4. Kỹ thuật tạo mô hình Quạt

a. Mô hình phân cấp



b. Các mô tả vắn tắt cho việc tạo mô hình Quạt

//Quạt

mat4 Canhquat\_view;

mat4 Quat\_view;

void Quat\_chandoc(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(0.0, 0.5 \* y, 0) \* Scale(x, y, z);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Quat\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void Quat\_canh(GLint g, GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = RotateY(g) \* Translate(x, y, z) \* Scale(0.5, 0.01, 0.2);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Quat\_view \* Canhquat\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void Quat\_de()

{

mat4 m = Translate(0, 0.2, 0) \* Scale(0.05, 0.02, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Quat\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void Quat\_chan()

{

Quat\_chandoc(0.03, 0.2, 0.03);

Quat\_de();

}

void Quat\_canhcc()

{

Quat\_canh(90, -0.5 \* 0.5, 0.0, 0.0);

Quat\_canh(-30, -0.5 \* 0.5, 0.0, 0.0);

Quat\_canh(-150, -0.5 \* 0.5, 0.0, 0.0);

}

GLfloat theta\_Quat[] = { 0, 0, 0, 0, 0, 0, 0, 0 };

void Quat(GLfloat x, GLfloat y,GLfloat z)

{

MakeColor(0, 0.6, 0, 1);

Quat\_view = Translate(x,y,z) \* Scale(0.3, 0.3, 0.3) \* RotateY(theta\_Quat[1]) \* RotateX(theta\_Quat[2]) \* Translate(0, theta\_Quat[5], 0) \* Translate(theta\_Quat[6], 0, 0) \* Translate(0, 0, theta\_Quat[7]) \* RotateY(180) \* Scale(2, 2, 2);

Canhquat\_view = Translate(theta\_Quat[6], theta\_Quat[5], theta\_Quat[7]) \* RotateY(theta\_Quat[0]);

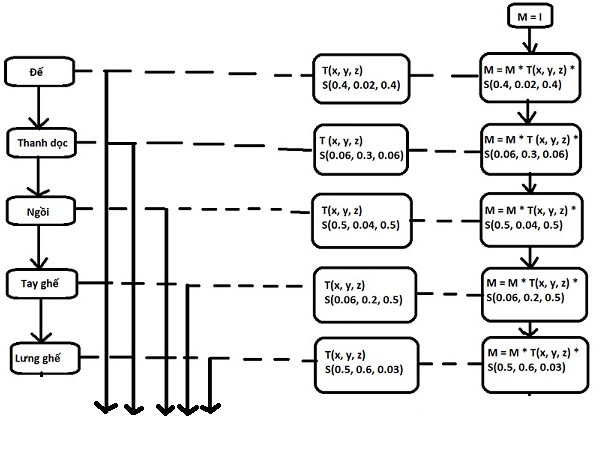
Quat\_chan();

Quat\_canhcc();

}

## 2.5. Kỹ thuật tạo mô hình Ghế giáo viên

a. Mô hình phân cấp



b. Các mô tả vắn tắt cho việc tạo mô hình Ghế giáo viên

//Ghế

mat4 Ghe\_view;

mat4 Ghetren\_view;

mat4 Tuadau\_view;

//ghexoay

void de(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z)\*Scale(0.4, 0.02, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ghe\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void thanhdoc(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z)\*Scale(0.06, 0.3, 0.06);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ghe\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void ngoi(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.5, 0.04, 0.5);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ghetren\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void tayghe(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.06, 0.2, 0.5);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ghetren\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void lungghe(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.5, 0.6, 0.03);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ghetren\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void tuadau(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.2, 0.1, 0.05);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Tuadau\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void thanhtua(GLfloat x, GLfloat y, GLfloat z)

{

mat4 m = Translate(x, y, z) \* Scale(0.08, 0.2, 0.01);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Tuadau\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void ghe\_duoi()

{

de(0, -0.5 \* 0.02 - 0.3, 0);

thanhdoc(0, -0.5 \* 0.3, 0);

}

void ghe\_tren()

{

ngoi(0, 0.04\*0.5, 0);

tayghe(-0.5\*(0.5 - 0.05), 0.5\*0.2, 0);

tayghe(0.5 \* (0.5 - 0.05), 0.5 \* 0.2, 0);

lungghe(0, 0.5 \* 0.6, -0.5\*(0.5 - 0.02));

}

//Than

void Ghe()

{

MakeColor(0.2, 0.8, 0.5, 1.0);

Ghe\_view = Translate(-(1.5 - 0.45) + 0.05, -0.4 + 0.2, -1.2 + 0.025 + 0.05 - 0.35) \* Translate(xghe, 0, zghe) \* RotateY(0) \* Scale(0.5, 0.5, 0.5);

ghe\_duoi();

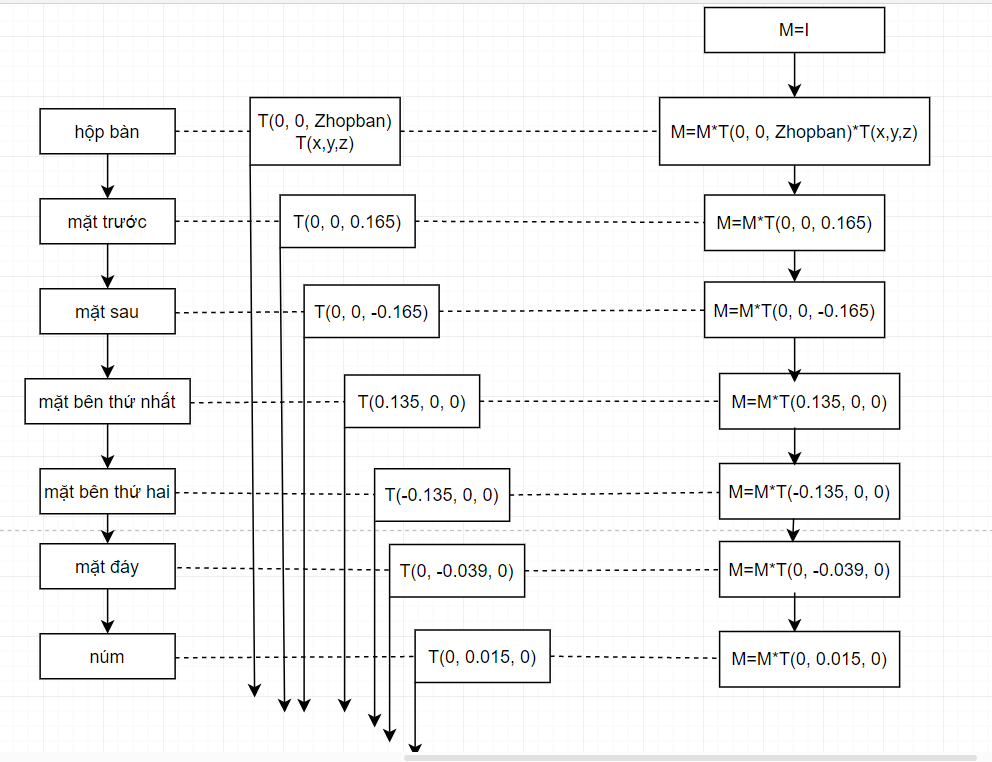
Ghetren\_view = Ghe\_view \* RotateY(rghe);

ghe\_tren();

}

## 2.6. Kỹ thuật tạo mô hình Ngăn kéo

a. Mô hình phân cấp



b. Các mô tả vắn tắt cho việc tạo mô hình Ngăn kéo

mat4 HopBan\_view;

//Hop ban

void HopBan(GLfloat x, GLfloat y, GLfloat z)

{

HopBan\_view = Translate(0.0, 0.0, zhopban) \* Translate(x, y, z);

MakeColor(0, 0.5, 0, 1);

//Mat Truoc

mat4 m = Translate(0.0, 0.0, 0.165) \* Scale(0.3, 0.15, 0.03);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* HopBan\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat sau

m = Translate(0.0, 0.0, -0.165) \* Scale(0.3, 0.1, 0.03);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* HopBan\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat ben

m = Translate(0.135, 0.0, 0.0) \* Scale(0.03, 0.1, 0.3);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* HopBan\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat ben

m = Translate(-0.135, 0.0, 0.0) \* Scale(0.03, 0.1, 0.3);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* HopBan\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat day

MakeColor(0.5, 0.2, 0, 1);

m = Translate(0.0, -0.039, 0.0) \* Scale(0.3, 0.02, 0.3);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* HopBan\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Num

MakeColor(1.0, 0.2, 0.2, 1);

m = Translate(0.0, 0.015, 0.18) \* Scale(0.05, 0.02, 0.02);

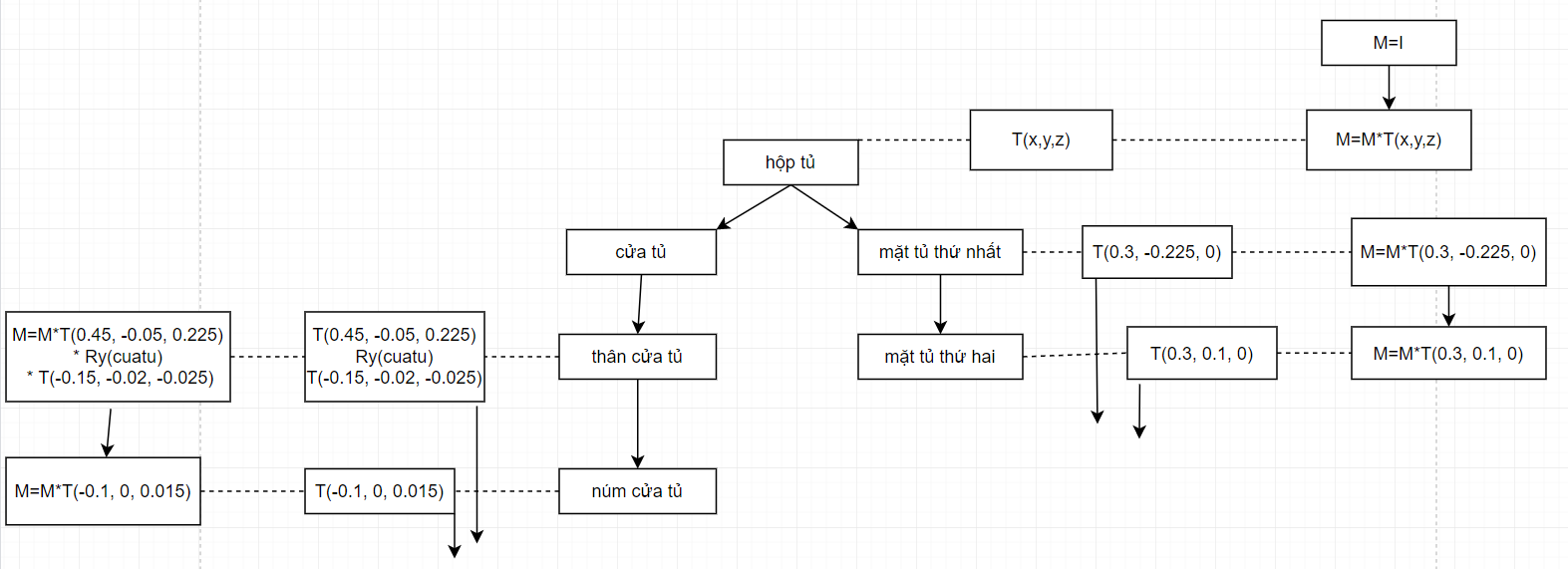
glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* HopBan\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

## 2.7. Kỹ thuật tạo mô hình Tủ bàn giáo viên

a. Mô hình phân cấp



b. Các mô tả vắn tắt cho việc tạo mô hình Tủ bàn giáo viên

mat4 CuaTu\_view;

//Cua tu

void CuaTu()

{

//Than cua

MakeColor(0, 0.5, 0, 1);

CuaTu\_view = Translate(0.45, -0.05, 0.225) \* RotateY(cuatu) \* Translate(-0.15, -0.02, -0.025);

mat4 m = Scale(0.3, 0.3, 0.03);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* CuaTu\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Num cua

MakeColor(1.0, 0.2, 0.2, 1);

m = Translate(-0.1, 0, 0.015) \* Scale(0.03, 0.07, 0.02);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* CuaTu\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

## 2.8. Kỹ thuật điều chỉnh camera

mat4 projection;

GLuint projection\_loc;

mat4 view;

GLuint view\_loc;

GLfloat px = 3, py = 0.0, pz = 0.0;

vec3 viewer\_pos(0, 0, 2);

point4 eye(px \* sin(py) \* cos(pz), px \* sin(py) \* sin(pz), px \* cos(py), 1.0);

vec4 at(0, 0, 1, 1);

vec4 up(0, 1, 0, 1);

view = LookAt(eye, at, up);

glUniformMatrix4fv(view\_loc, 1, GL\_TRUE, view);

projection = Frustum(-0.5, 0.5, -0.5, 0.5, 1, 50);

glUniformMatrix4fv(projection\_loc, 1, GL\_TRUE, projection);

## 2.9. Kỹ thuật chiếu sáng

uniform vec4 AmbientProduct, DiffuseProduct, SpecularProduct;

uniform vec4 LightPosition;

uniform float Shininess;

uniform mat4 Model;

uniform mat4 View;

uniform mat4 Projection;

void main()

{

//Mo hinh chieu sang Blinn - Phong (Phong sua doi)

// Transform vertex position into eye coordinates

vec3 pos = (View \*Model\* vPosition).xyz;

vec3 L = normalize( LightPosition.xyz - pos );

vec3 E = normalize( -pos );

vec3 H = normalize(L+E);

// Transform vertex normal into eye coordinates

vec3 N = normalize( View \* Model\* vec4(vNormal, 0.0) ).xyz;

// Compute terms in the illumination equation

vec4 ambient = AmbientProduct;

float Kd = max( dot(L, N), 0.0 );

vec4 diffuse = Kd\*DiffuseProduct;

float Ks = pow( max(dot(N, H), 0.0), Shininess );

vec4 specular = Ks \* SpecularProduct;

if( dot(L, N) < 0.0 ) specular = vec4(0.0, 0.0, 0.0, 1.0);

color = ambient + diffuse + specular;

color.a = 1.0;

gl\_Position = Projection \* View \* Model \* vPosition/vPosition.w;

}//

## 2.10. Kỹ thuật tạo mô hình Bàn giáo viên

* Các mô tả vắn tắt cho việc tạo mô hình Bàn giáo viên

void Ban()

{

Ban\_view = Translate(-(1.5 - 0.45) + 0.05, -0.4+0.2, -1.2 + 0.025 + 0.05 + 0.1) \* Scale(0.8, 0.8, 0.7) \* RotateY(180);

MakeColor(0.5, 0.5, 0.9, 1);

Chan(-0.375, 0.0, 0.0);

Chan(0.125, 0.0, 0.0);

Chan(0.475, 0.0, 0.0);

//Mat sau

mat4 m = Translate(0.05, 0.0, -0.22) \* Scale(0.9, 0.5, 0.03);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat Ban

m = Translate(0.05, 0.275, 0.0) \* Scale(0.9, 0.05, 0.55);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat tu

m = Translate(0.3, -0.225, 0.0) \* Scale(0.3, 0.05, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

//Mat tu

m = Translate(0.3, 0.1, 0.0) \* Scale(0.3, 0.05, 0.4);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Ban\_view \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

CuaTu();

HopBan(0.3, 0.17, 0.03);

}

## 2.11. Kỹ thuật tạo mô hình Bàn học sinh

* Các mô tả vắn tắt cho việc tạo mô hình Bàn học sinh

// Phần bàn ghế sinh viên

mat4 model\_table;

void matBan()

{

instance = Scale(0.6, 0.02, 0.2);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void nganBan() {

instance = Translate(0, -0.1, 0) \* Scale(0.6 - 0.04, 0.015, 0.2 - 0.04);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void chanBan(GLfloat x, GLfloat y, GLfloat z) {

instance = Translate(x, y, z) \* Scale(0.02, 0.4, 0.02);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void bonChanBan() {

chanBan(0.3 - 0.01, -0.2 - 0.015 / 2, 0.1 - 0.01);

chanBan(0.3 - 0.01, -0.2 - 0.015 / 2, -0.1 + 0.01);

chanBan(-(0.3 - 0.01), -0.2 - 0.015 / 2, -0.1 + 0.01);

chanBan(-(0.3 - 0.01), -0.2 - 0.015 / 2, 0.1 - 0.01);

}

void thanhNgang(GLfloat x, GLfloat y) {

instance = Translate(x, y, 0) \* Scale(0.015, 0.015, 0.2 - 0.04);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void haiThanhNgangBan() {

thanhNgang(0.3 - 0.015 / 2, -0.38 - 0.015 / 2);

thanhNgang(-0.3 + 0.015 / 2, -0.38 - 0.015 / 2);

}

void thanhNoiHaiThanhNgang() {

instance = Translate(0, -0.38 - 0.015 / 2, 0) \* Scale(0.6 - 0.015 \* 2, 0.015, 0.015);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void thanhNgangNoiBanGhe(GLfloat x, GLfloat y, GLfloat z) {

instance = Translate(x, y, z) \* Scale(0.015, 0.015, 0.20);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void haiThanhNoiBanGhe() {

thanhNgangNoiBanGhe(0.3 - 0.015 / 2, -0.38 - 0.015 / 2, 0.1 + 0.1);

thanhNgangNoiBanGhe(-0.3 + 0.015 / 2, -0.38 - 0.015 / 2, 0.1 + 0.1);

}

// cai ghe

void matGhe()

{

instance = Translate(0, -0.2, -0.12) \* Scale(0.6, 0.015, 0.2);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void chanGhe(GLfloat x, GLfloat y, GLfloat z) {

instance = Translate(x, y, z) \* Scale(0.02, 0.2, 0.02);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void thanhNgangCuaGhe(GLfloat x, GLfloat y)

{

instance = Translate(x, y, -0.1-0.02) \* Scale(0.015, 0.015, 0.2 - 0.04);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model\_table \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void haiThanhNgangGhe() {

thanhNgangCuaGhe(0.3 - 0.015 / 2, -0.38 - 0.015 / 2);

thanhNgangCuaGhe(-0.3 + 0.015 / 2, -0.38 - 0.015 / 2);

}

void bonChanGhe() {

chanGhe(0.3 - 0.01, -0.3 - 0.015 / 2, -0.12+0.1 - 0.01);

chanGhe(0.3 - 0.01, -0.3 - 0.015 / 2, -0.12 + -0.1 + 0.01);

chanGhe(-0.3 + 0.01, -0.3 - 0.015 / 2, -0.12 + 0.1 - 0.01);

chanGhe(-0.3 + 0.01, -0.3 - 0.015 / 2, -0.12 + -0.1 + 0.01);

}

void banSinhVien(GLfloat x, GLfloat y, GLfloat z)

{

model\_table = model \* Translate(x,y,z);

matBan();

nganBan();

bonChanBan();

haiThanhNgangBan();

thanhNoiHaiThanhNgang();

haiThanhNoiBanGhe();

model\_table = model\_table \* Translate(0, 0, 0.5);

bonChanGhe();

haiThanhNgangGhe();

matGhe();

}

## 2.12. Kỹ thuật tạo mô hình Bục giảng

* Các mô tả vắn tắt cho việc tạo mô hình Bục giảng

//Buc Giang

void buc()

{

MakeColor(0, 0.5, 0.3, 1);

mat4 m = Translate(-(3.05-2.75)/2, -0.5, -1.2+0.025) \* Scale(2.7, 0.2, 1);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* m);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

## 2.13. Kỹ thuật tạo mô hình Bảng

* Các mô tả vắn tắt cho việc tạo mô hình Bảng

//Bang

mat4 instance;

mat4 bang\_view, Chieudai\_view, Chieurong\_view;

void bang() {

MakeColor(0.2, 0.8, 0.5, 1.0);

instance = Translate(0,0,0) \* Scale(1.25, 0.5, 0.01);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \*bang\_view \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void vienChieuDai(GLfloat x, GLfloat y, GLfloat z) {

instance = Translate(x, y, z) \* Scale(1.25, 0.02, 0.015);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \*Chieudai\_view \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void haiVienChieuDai() {

MakeColor(1.0, 0.2, 0.2, 1);

vienChieuDai(0,0.24,0);

vienChieuDai(0,-0.24,0);

}

void vienChieuRong(GLfloat x, GLfloat y, GLfloat z) {

instance = Translate(x, y, z) \* Scale(0.02, 0.5, 0.015);

glUniformMatrix4fv(model\_loc, 1, GL\_TRUE, model \* Chieurong\_view \* instance);

glDrawArrays(GL\_TRIANGLES, 0, NumPoints);

}

void haiVienChieuRong() {

MakeColor(1.0, 0.2, 0.2, 1);

vienChieuRong((1.25/2), 0, 0);

vienChieuRong(-(1.25/2), 0, 0);

}

void bangFull() {

bang\_view = Translate(0, 0.1, -1.2 + 0.025 - 0.5);

bang();

Chieudai\_view = Translate(0, 0.1, -1.2 + 0.025 - 0.5);

haiVienChieuDai();

Chieurong\_view = Translate(0, 0.1, -1.2 + 0.025 - 0.5);

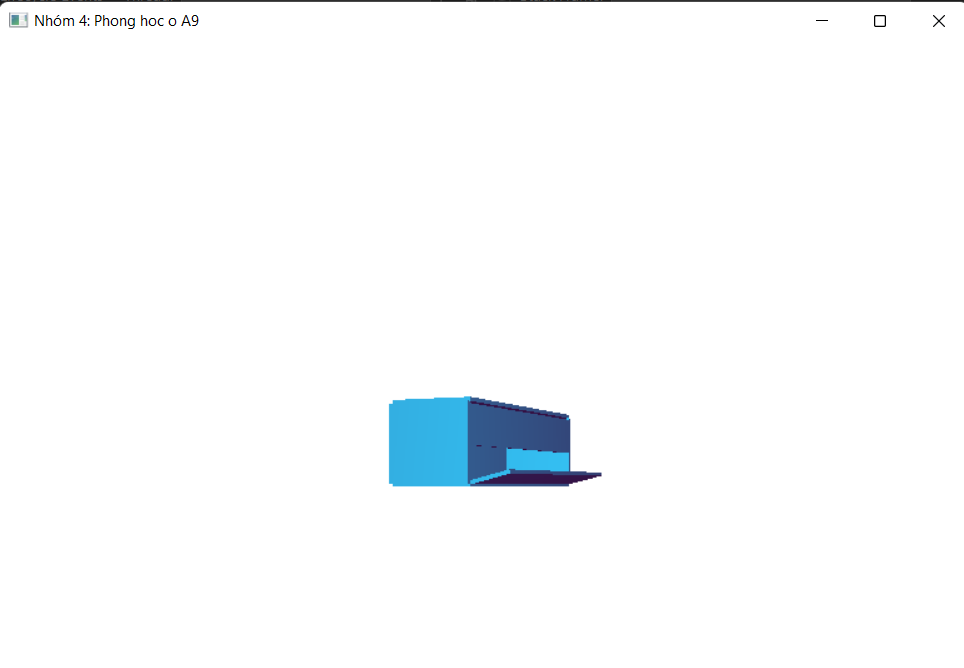
haiVienChieuRong();

}

# Chương 3 : Kết quả đạt được

## 3.1. Mô hình Điều hòa

1. Hình ảnh



1. Mô tả hoạt động

//Mở điều hoà

case 'F': rdieuhoa += 5;

if (rdieuhoa > 150) rdieuhoa = 0;

glutPostRedisplay();

break;

case 'f': rdieuhoa -= 5;

if (rdieuhoa < -150) rdieuhoa = 0;

glutPostRedisplay();

break;

## 3.2. Mô hình Quạt

1. Hình ảnh



1. Mô tả hoạt động

// Chuyển động của quạt

case 'm':

//Cánh quạt quay

theta\_Quat[0] += 15;

theta\_Quat[4] += 15;

if (theta\_Quat[0] > 360)

theta\_Quat[0] -= 360;

if (theta\_Quat[4] > 360)

theta\_Quat[4] -= 360;

glutPostRedisplay();

break;

case 'M':

//Cánh quạt quay

theta\_Quat[0] -= 15;

theta\_Quat[4] -= 15;

if (theta\_Quat[0] < -360)

theta\_Quat[0] = 360;

if (theta\_Quat[4] < -360)

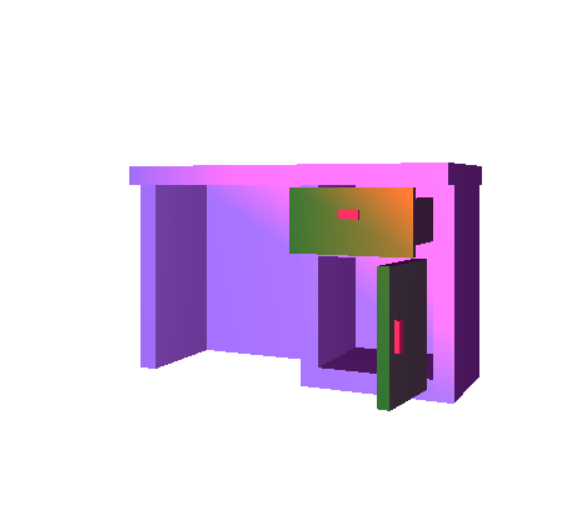
theta\_Quat[4] = 360;

glutPostRedisplay();

break;

## 3.3. Mô hình Bàn giáo viên có đủ ngăn kéo, tủ bàn

1. Hình ảnh



1. Mô tả hoạt động

//Tủ bàn giáo viên

case 'o': //Mở cửa tủ

if (cuatu < 160)

cuatu += 5;

glutPostRedisplay();

break;

case 'i': //Đóng cửa tủ

if (cuatu > 0)

cuatu -= 5;

glutPostRedisplay();

break;

//Ngăn kéo

case 'r': //kéo hộp bàn ra

if (zhopban < 0.25)

zhopban += 0.015;

glutPostRedisplay();

break;

case 't': //kéo hộp bàn vào

if (zhopban >= 0.015)

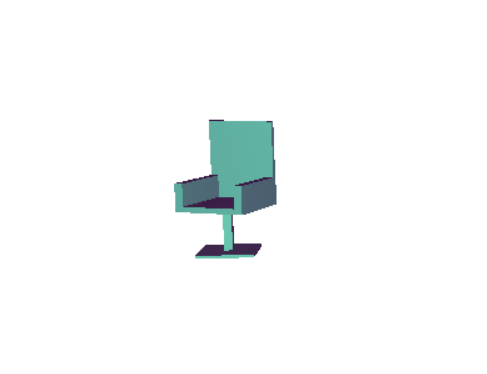
zhopban -= 0.015;

glutPostRedisplay();

break;

## 3.4. Mô hình Ghế giáo viên

1. Hình ảnh



1. Mô tả hoạt động

//Ghế

case 'w': //Kéo ghế vào

if (zghe < 0.35 && xghe <= 0.03)

zghe += 0.01;

glutPostRedisplay();

break;

case 's': //Kéo ghế ra

if (zghe > -0.1 && xghe <= 0.03)

zghe -= 0.01;

glutPostRedisplay();

break;

case 'd': //sang phải

if (xghe < 0.18 && zghe <= 0.1)

xghe += 0.01;

glutPostRedisplay();

break;

case 'a': //Sang trái

if (xghe > -0.05 && zghe <= 0.1)

xghe -= 0.01;

glutPostRedisplay();

break;

case 'e':

rghe += 5;

if (rghe > 360)

rghe -= 360;

glutPostRedisplay();

break;

case 'E':

rghe -= 5;

if (rghe > 360)

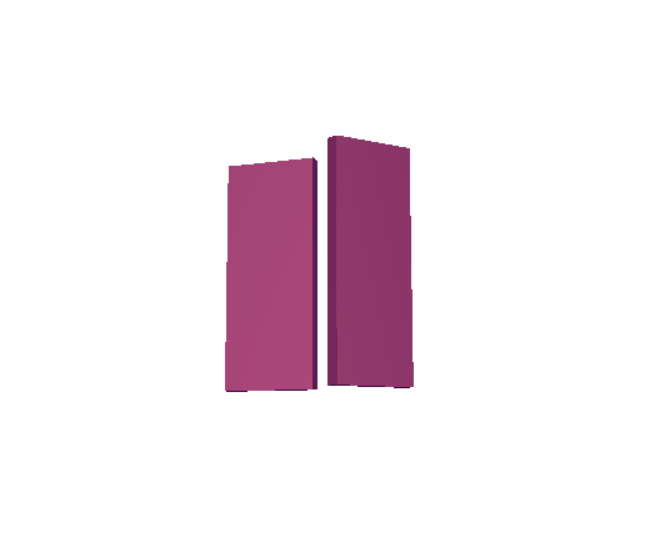
rghe = 0;

glutPostRedisplay();

break;

## 3.5. Mô hình Cửa sổ

1. Hình ảnh



1. Mô tả hoạt động

//Cua so

case 'h':

if (trai < 160)

trai += 5;

glutPostRedisplay();

break;

case 'H':

if (trai > 0)

trai -= 5;

glutPostRedisplay();

break;

case 'J':

if (phai < 0)

phai += 5;

glutPostRedisplay();

break;

case 'j':

if (phai > -160)

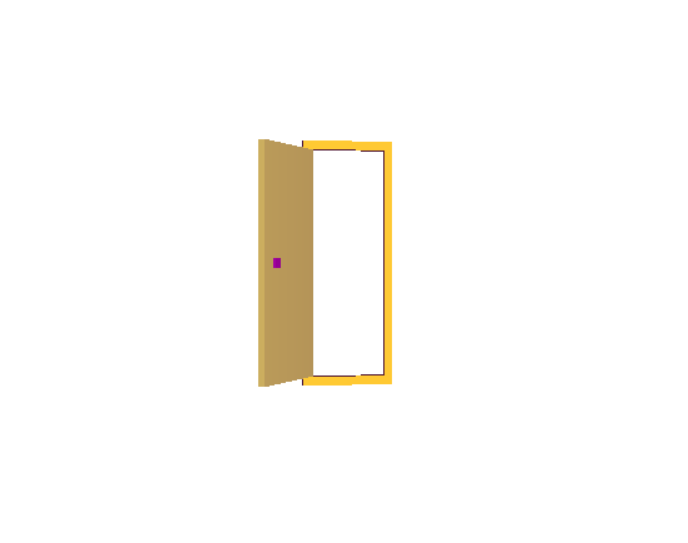
phai -= 5;

glutPostRedisplay();

break;

## 3.6. Mô hình Cửa ra vào

1. Hình ảnh



1. Mô tả hoạt động

//Cua ra vao

case 'C':

if (cua < 180)

cua += 5;

glutPostRedisplay();

break;

case 'c':

if (cua > 50)

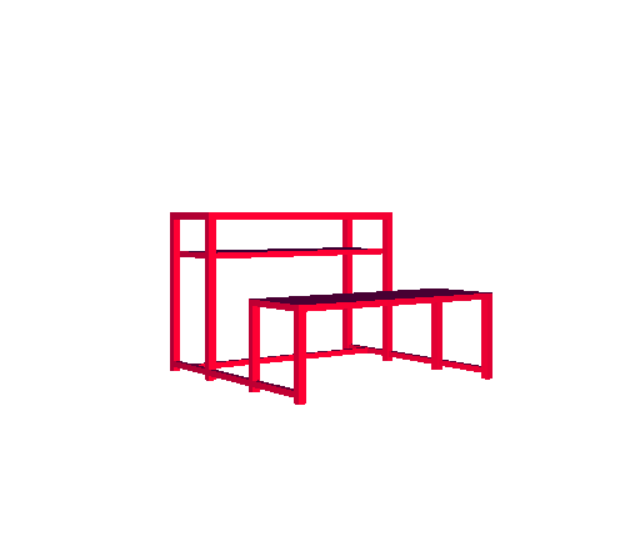
cua -= 5;

glutPostRedisplay();

break;

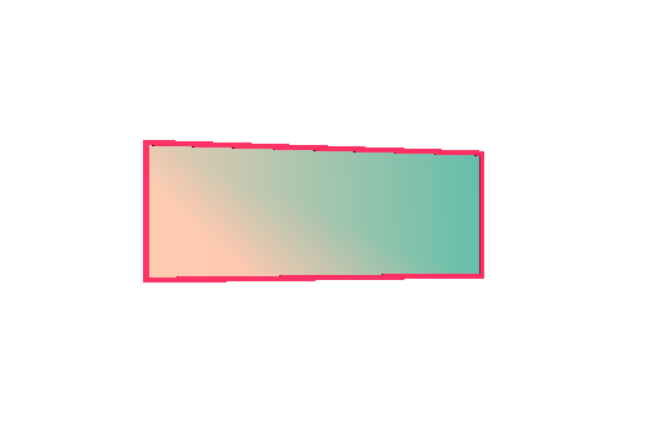
## 3.7. Mô hình Bàn học sinh

1. Hình ảnh



## 3.8. Mô hình Bảng

1. Hình ảnh



## 3.9. Điều khiển camera

1. Mô tả hoạt động

//View

case '1':

//if (px < 2)

px += 0.1;

glutPostRedisplay();

break;

case '2':

//if (px > -2)

px -= 0.1;

glutPostRedisplay();

break;

case '3':

//if (py < 2)

py += 0.1;

glutPostRedisplay();

break;

case '4':

//if (py > 0)

py -= 0.1;

glutPostRedisplay();

break;

case '5':

//if (pz < 4)

pz += 0.1;

glutPostRedisplay();

break;

case '6':

//if (pz > 0)

pz -= 0.1;

glutPostRedisplay();

break;

case ' ': // Đưa về góc nhìn mặc định

px = 4.0;

py = 0.0;

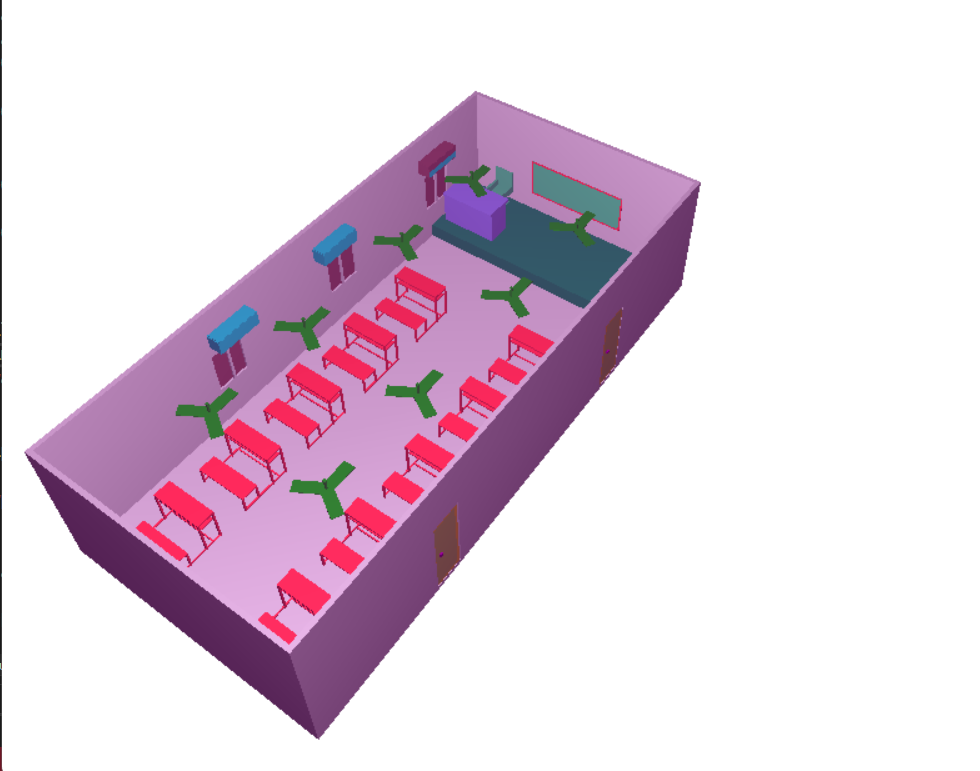
pz = 0.0;

glutPostRedisplay();

break;

## 3.10. Mô hình mô phỏng toàn bộ phòng học lý thuyết thuộc nhà A9

1. Hình ảnh



1. Mô tả hoạt động

//Xoay phong

case 'x':

xphong += 5;

if (xphong > 360)

xphong -= 360;

glutPostRedisplay();

break;

case 'X':

// một số lệnh

xphong -= 5;

if (xphong > 360)

xphong -= 360;

glutPostRedisplay();

break;

case 'y':

// một số lệnh

yphong += 5;

if (yphong > 360)

yphong -= 360;

glutPostRedisplay();

break;

case 'Y':

// một số lệnh

yphong -= 5;

if (yphong > 360)

yphong -= 360;

glutPostRedisplay();

break;

case 'z':

// một số lệnh

zphong += 5;

if (zphong > 360)

zphong -= 360;

glutPostRedisplay();

break;

case 'Z':

// một số lệnh

zphong -= 5;

if (zphong > 360)

zphong = 0;

glutPostRedisplay();

break;

# KẾT LUẬN

Qua việc thực hiện nghiên cứu đề tài “Mô phỏng một phòng học lý thuyết thuộc nhà A9 của trường ĐH Công Nghiệp Hà Nội”, nhóm chúng em đã được biết thêm rất nhiều về môn đồ họa máy tính. Bên cạnh đó, việc làm nghiên cứu giúp chúng em đoàn kết hơn, rèn luyện cho chúng em kỹ năng làm việc nhóm.

Trong quá trình thực hiện đề tài có rất nhiều ý tưởng hay, độc đáo. Nhưng do kiến thức của chúng em hạn hẹp và thời gian không cho phép nên chúng em chưa thể thực hiện được những ý tưởng đó. Tuy nhiên chúng em đã cố gắng để xây dựng một chương trình hoàn chỉnh nhất, đẹp nhất. Trong quá trình xây dựng chương trình, nhóm chúng em khó tránh khỏi những sai sót. Vì vậy em rất mong rằng cô và các bạn cùng góp ý với nhóm em để có thể hoàn thành sản phẩm một cách hoàn chỉnh nhất.

Chúng em xin cảm ơn cô Vũ Minh Yến đã tận tình giảng dạy chúng em trong môn Đồ họa máy tính. Giúp đỡ chúng em trong quá trình nghiên cứu đề tài. Chia sẻ những tài liệu hay về thư viện OpenGl cũng như các kĩ năng lập trình.

Chúng em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

[1] GIÁO TRÌNH ĐỒ HỌA MÁY TÍNH

[2] Các tài liệu học tập của bộ môn trên lớp học Class

[3] Các tài liệu khác trên Internet.