**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**\*\*\***



**BÁO CÁO BÀI TẬP**

**ĐỒ HỌA MÁY TÍNH**

**ĐỀ TÀI**: Viết chương trình mô phỏng phép tịnh tiến, phép biến đổi tỉ lệ, phép quay trục, phóng to thu nhỏ, biến dạng của một hình bất kỳ trong không gian 3 chiều

**Thành viên**: nhóm 7

**Nguyễn Thị Thùy Linh**

**Phạm Văn Bách**

**Bùi Văn Thắng**

**GIẢNG VIÊN:** Nguyễn Thị Cẩm Ngoan

Hà Nội, ngày 05 tháng 02 năm 2013

**Chi tiết công việc :**

|  |  |  |
| --- | --- | --- |
| **Mã Sinh Viên** | **Họ Tên** | **Phân Công** |
| **0641360154** | **Bùi Văn Thắng** | **Phân tích các phép toán, viết trương trình mô phỏng phép quay trụ, phép phóng to thu nhỏ** |
| **0641360185** | **Phạm Văn Bách** | **Phân tích các phép toán, Làm báo cáo, viết trương trình cho phép tịnh tiến và phép biến dạng,** |
| **0641360224** | **Nguyễn Thị Thùy Linh** | **Phân tích các phép toán, viết trương trình cho thuất toán biến đổi tỉ lệ, làm báo cáo** |

**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 cho phép tạo, hiển thị và điều khiển hình ảnh trên màn hình máy tính. Đồ họa máy tính liên quan ít nhiều đến một số lĩnh vực như đại số, hình học giải tích và hình học họa hình, quang học…… và kĩ thuật máy tính và đặc biệt là chế tạo phần cứng ( các loại màn hình, các thiết bị nhập xuất, các vi mạch đồ họa).

Nghiên cứu các phép biến đổi hình học trong không gian thực hai chiều là một trong những nội dung quan trọng.

Qua thời gian tìm hiểu nhóm quyết định trọn đề tài Viết chương trình mô phỏng sự va đập của một trái bóng trên màn hình. (Nếu gặp đường biên màn hình thì đổi hướng chuyển động theo định luật phản xạ ánh sang) nhằm mô phỏng cơ chế va đập của quả bóng trong không gian thực hai chiều.

Do quá trình tìm hiểu còn nhiều thiếu sót nên chương trình còn nhiều hạn chế, nhóm thực hiện rất mong nhận được ý kiến đóng góp từ phía cô giáo và các bạn để phần mềm được hoàn thiện hơn.

**MỤC LỤC**

CHƯƠNG I: KHẢO SÁT 5

I. Mục đích nghiên cứu 5

II. Đối Tượng nghiên cứu 5

III. Khảo sát 5

IV.Hướng giải quyết 5

CHƯƠNG II: PHÂN TÍCH 6

I. Phép tịnh tiến 6

II. Phép biến đổi tỷ lệ 6

III. Phép biến dạng 7

IV. Phép biến dạng 8

V. Phép phóng to thu nhỏ 13

CHƯƠNG III: CÀI ĐẶT CHƯƠNG TRÌNH 13

III. Triển khai ứng dụng 13

- Phép tịnh tiến 13

-Phép phóng to thu nhỏ 13

-Phép quay trục 14

-Phép biến đổi tỷ lệ 14

**3.2** Cài đặt chương trình 15

TỰ ĐÁNH GIÁ 21

KẾT LUẬN 21

**CHƯƠNG I: KHẢO SÁT**

**I. Mục đích nghiên cứu**

Tạo ra được hình có nhiều tính năng trong không gian ba chiều như tịnh tiến, phóng to thu nhỏ, quay quanh trục, đối xứng ... Giúp chúng ta có thể hiểu và nhìn nhận tốt hơn về những phép chiếu đó.

**II. Đối tượng nghiên cứu**

Hướng đến tất cả mọi người sử dụng máy tính, đặc biệt là những người tuổi nhỏ đang tuổi học hỏi. Giúp họ thấy thú vị và hứng thú khi được nhìn thấy các vật thể ba chiều trong không gian ảo. Tạo điều kiện thuận lợi cho việc nghiên cứu nhất là trong học tâp.

**III. Khảo sát**

# Qua khảo sát trên các máy tính sử dụng hệ điều hành Windows, trên win xp thì không có hiệu ứng màn hình chờ quả bóng có sự va đập trên màng hình, còn win 7,8 thì cũng đã có nhưng xuất hiện nhiều quả bóng và một quả không tồn tại mãi.

Khảo sát cũng cho thấy trên hệ điều hành Windows 7,8 có hiệu ứng sự va đập nhưng có sự xuất hiện của nhiều quả bóng. Trên hai hệ điều hành này còn có thêm hiệu ứng gió thổi nên sẽ không thấy sự ổn định chuyển động của quả bóng.

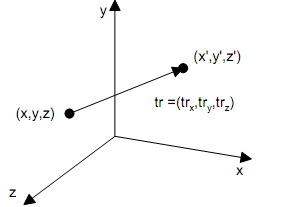
**IV. Hướng giải quyết**

Dự vào thư viện đồ họa chúng ta sẽ tạo ra các điểm trên trục tọa độ ảo trong không gian để tạo lên vật thể. Từ đó sẽ sử dụng các thuật toán cơ bản như tịnh tiến phóng to thu nhỏ lên vật thể đó, và tạo các hiệu ứng màu trên đó.

**CHƯƠNG II: PHÂN TÍCH**

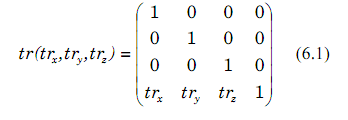
**Các phép biến hình cơ bản**

**I. Phép tịnh tiến**



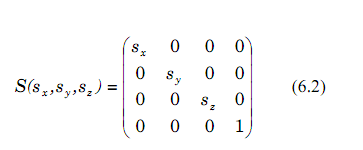
Vector tịnh tiến trong phép biến đổi ba chiều có một tác động rất trực quan.

Mỗi điểm được tính dịch đị một khoảng định trước la tx, ty,tz.. Ta có ma trận cho mỗi điểm M là như sau.

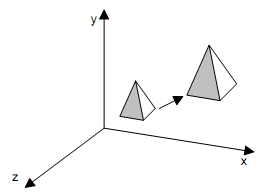


**II. Phép biến đổi tỉ lệ**

phép biến đổi tỷ lệ trong không gian ba chiều là một phép biến đổi mở rộng của phép biến đổi trong không gian hai chiều.



Trong đó các hằng số sx,sy,sz là các hằng số tỉ lệ với trục ox, oy, oz.



Như hình ta thấy phép biến đổi tỷ lệ cho ta thấy đối tượng được phông to lên và các điểm được di chuyển ra xa tọa độ gốc.

Khi cả ba hệ số sx, sy, sz bằng nhau ta được phép biến đổi đồng dạng.

Trong phép biến đổi gốc O chính là ảnh của nó, ta nói O chính là điểm bất động của S. Hay O chính là tâm của phép biến đổi.

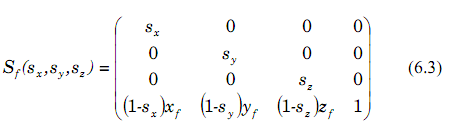
Tổng quát hơn ta có thể mô tả một phép biến đổi tỷ lệ thêo một tâm F bất kì bằng một dãy ba phép biến đổi sau.

-Tịnh tiến điểm bất động về tọa độ gốc.

-Thực hiện phép biến đổi tỷ lệ theo công thức ở hình 6.2.

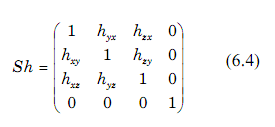
-Tịnh tiến ngược điểm bất động từ tọa độ gốc về điểm ban đầu.

Như vậy, ta hợp nhất được ba bước biến đổi ta được ma trận biến đổi tỷ lệ cho một điểm bất kì theo hệ số x,y,z là.



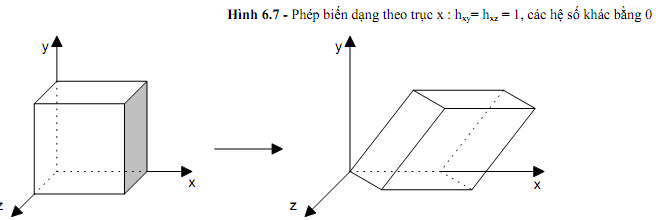
**III. Phép biến dạng**

Biến dạng theo trục nào của hệ tọa độ cũng bị ảnh hưởng từ hai trục còn lại. Ta có ma trận của phép biến dạnh như sau.

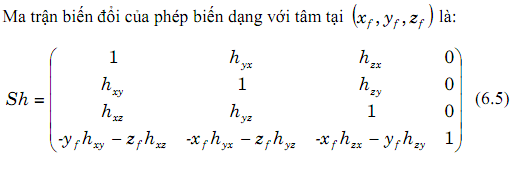


Ta có mối quan hệ Qx vớ P : Qx = Px +hxyPy + hxzPz

Ở đây có thể hiểu hxy là lượng mà tọa độ y của P tác động lên tọa độ x của Q



Tương tự trong phép biến đổi tỷ lệ, phép biến dạng SH trong hình 6.4 cũng là điểm bất động tại gốc tọa độ O. Ta cũng có thể xây dựng được phép biến dạng trong một trường hợp bấy kỳ là :



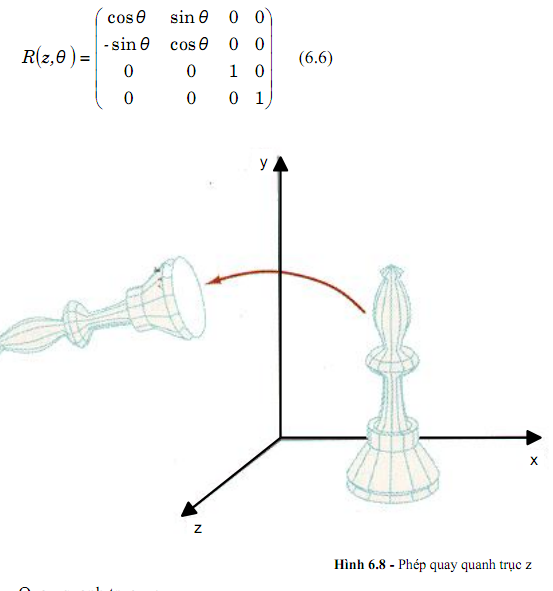
**IV. Phép quay trục**

4.1 Phép quay quanh một trục tọa độ

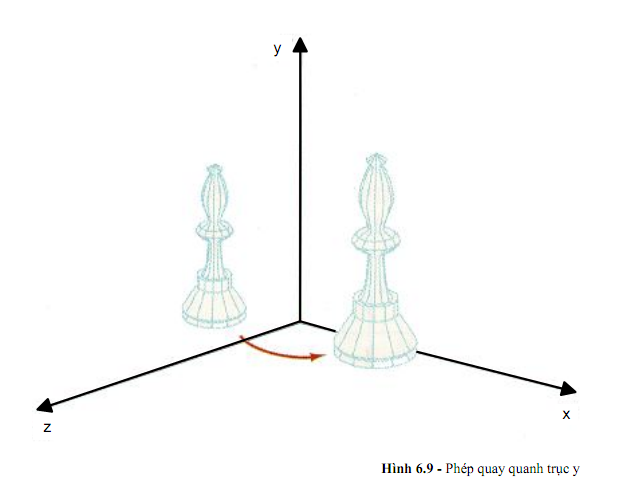
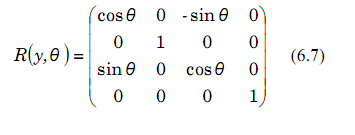
Khác với phép quay trong hai chiều quanh một điểm bất kì, trong ba chiều ta có phép quay quanh một trục tọa độ. Ở đây ta sử dụng hệ tọa độ theo quy ước bàn tay phải và quy định chiều quay dƣơng là ngược chiều kim

đồng hồ. Ta có các ma trận biểu diễn các phép quay quanh trục x, y, z một góc 0 lần lƣợt là R(z, 0), R(y,0), R(x, 0):

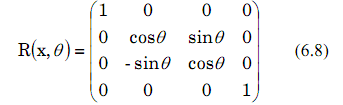
Quay quanh trục z :



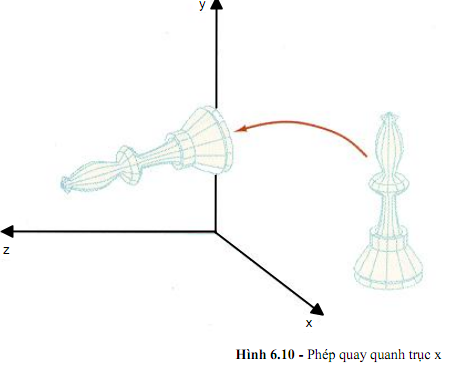
Quay quanh trục oy:



Quay quanh trục oz



Nhận xét rằng các giá trị nằm trên dòng và cột tƣơng ứng với trục x trong ma trận R(x,0) sẽ có giá trị là 0 ngoại trừ giá trị nằm trên đường chéo chính là 1. Điều này đảm bảo cho tọa độ x của các điểm là không bị thay đổi qua phép biến đổi. Nhận xét này cũng tương tự cho trường hợp các ma trận còn lại.



Ghi chú:

Các định nghĩa về chiều quay được dùng chung cho cả hệ tọa độ theo quy ước bàn tay phải và bàn tay trái. Cụ thể chiều dương được định nghĩa như sau:

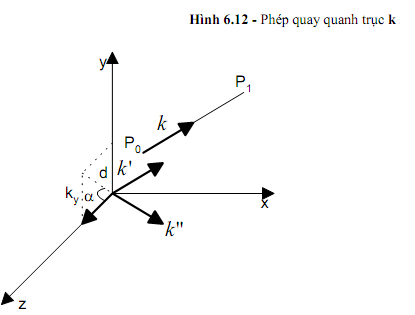
Quay quanh trục x: từ trục dương y đến trục dương z.

Quay quanh trục y: từ trục dương z đến trục dương x.

Quay quanh trục z: từ trục dương x đến trục dương y.

**4.2 phép tịnh tiến quay quanh một trục bất kỳ**

Giả sử trục quay đi qua hai điểm P0, P1 nào đó với phương trình biêu diễn bởi vector đơn vị k. Quay điểm (x, y, z) quanh trục k theo một góc nào đó nó sẽ biến thành điểm có tọa độ (x’,y’ z’)(như hình 6.12).



Để thực hiện phép quay quanh k một góc a, ta có thể thực hiện một chuỗi các thao tác sau:

Tịnh tiến trục k về gốc tọa độ: tr(-P0 ) (thành trục k').

Quay quanh trục x một góc để đặt trục k' nằm trên mặt phẳng Oxz: rot(x,a) (thành trục k”).

Quay quanh trục y góc để đưa trục k” về trục z: rot(y,-a).

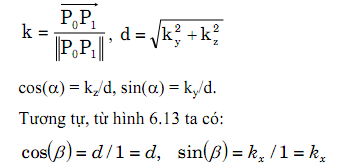
Thực hiện phép quay quanh trục z một góc a: rot(z,a).

Thực hiện chuỗi các phép biến đổi ngược lại quá trình trên.

Góc quay a được xác định dựa trên chiếu của k' lên mặt phẳng yz. Ta không cần tính a cụ thể. Thay vào đó ta tính sin(a)

và cos(a) một cách trực tiếp.

Từ hình 6.12 ta có:

****

Như vậy, phép quay quanh một trục P0P1 bất kì một góc a, rot(P0P1,a), có thể được phân rã thành chuỗi các biến đổi cơ sở sau:

tr(-P0 ) rot(x,a) rot(y, -b) rot(z, a) rot(y, a) rot(x, -a) tr(P0 )

**V. Phép phóng to thu nhỏ**

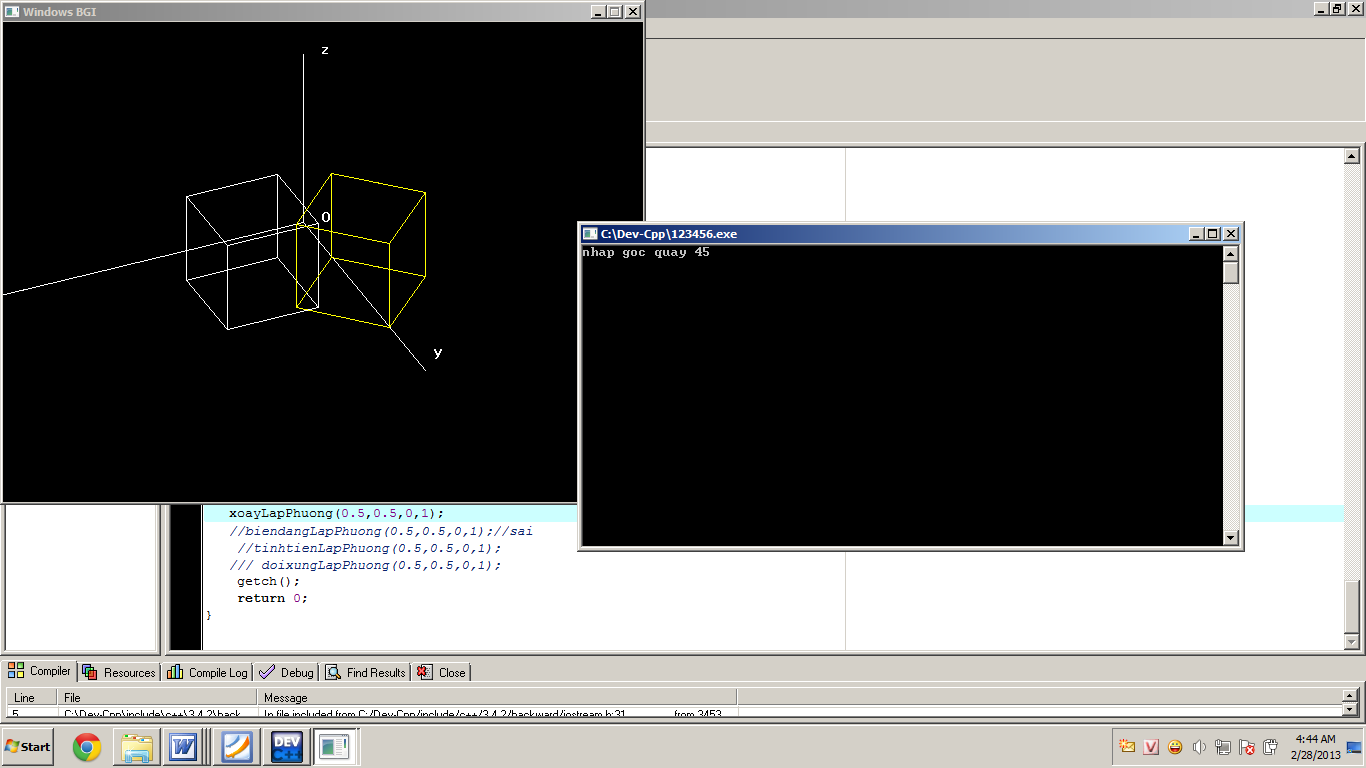
Phép phóng to thu nhỏ tương thự như phép biến đổi tỉ lệ theo hệ số riêng x,y,z bằng nhau tạo ra phép biến đổi tỷ lệ đồng dạng. Công thức như phép biến đổi tỷ lệ như trình bày ở trên.

# CHƯƠNG III: CÀI ĐẶT CHƯƠNG TRÌNH

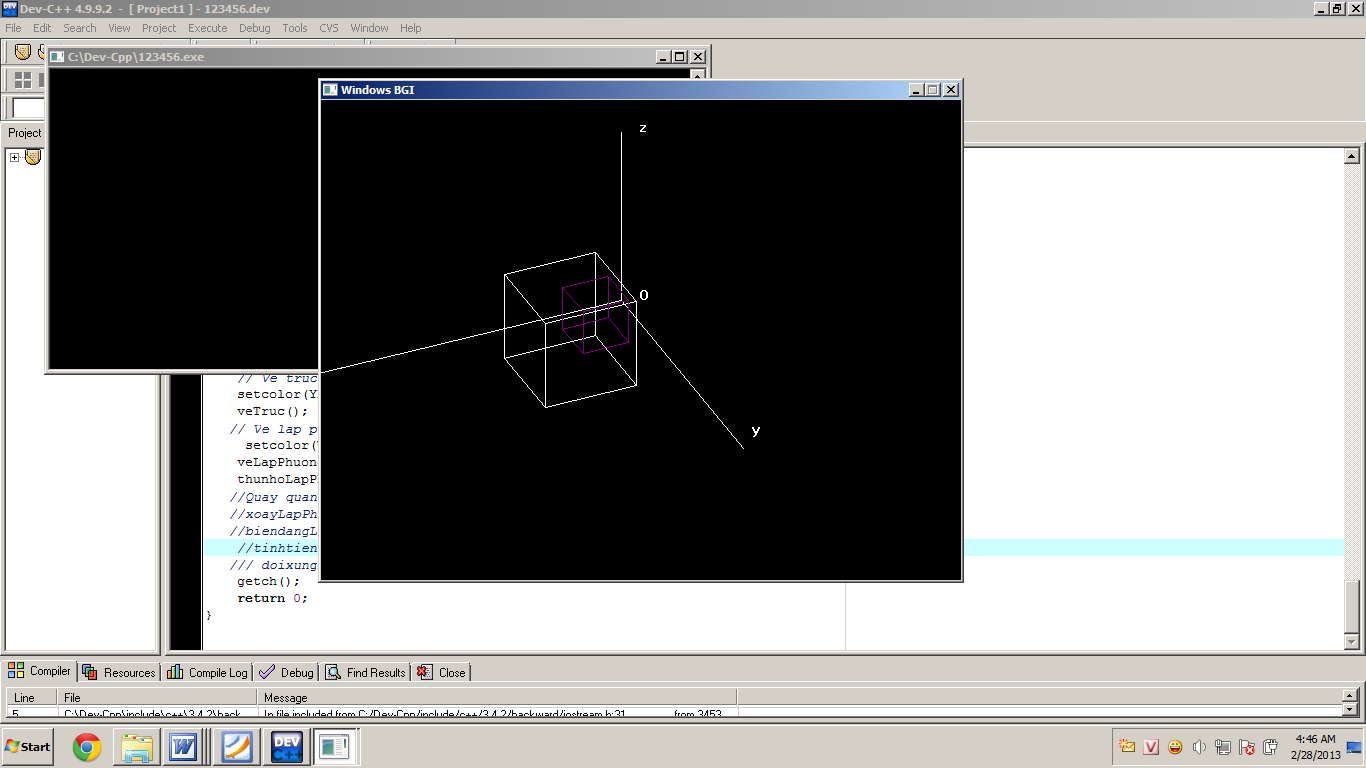
**III. Triển khai ứng dụng**

**Giao diện chương trình**

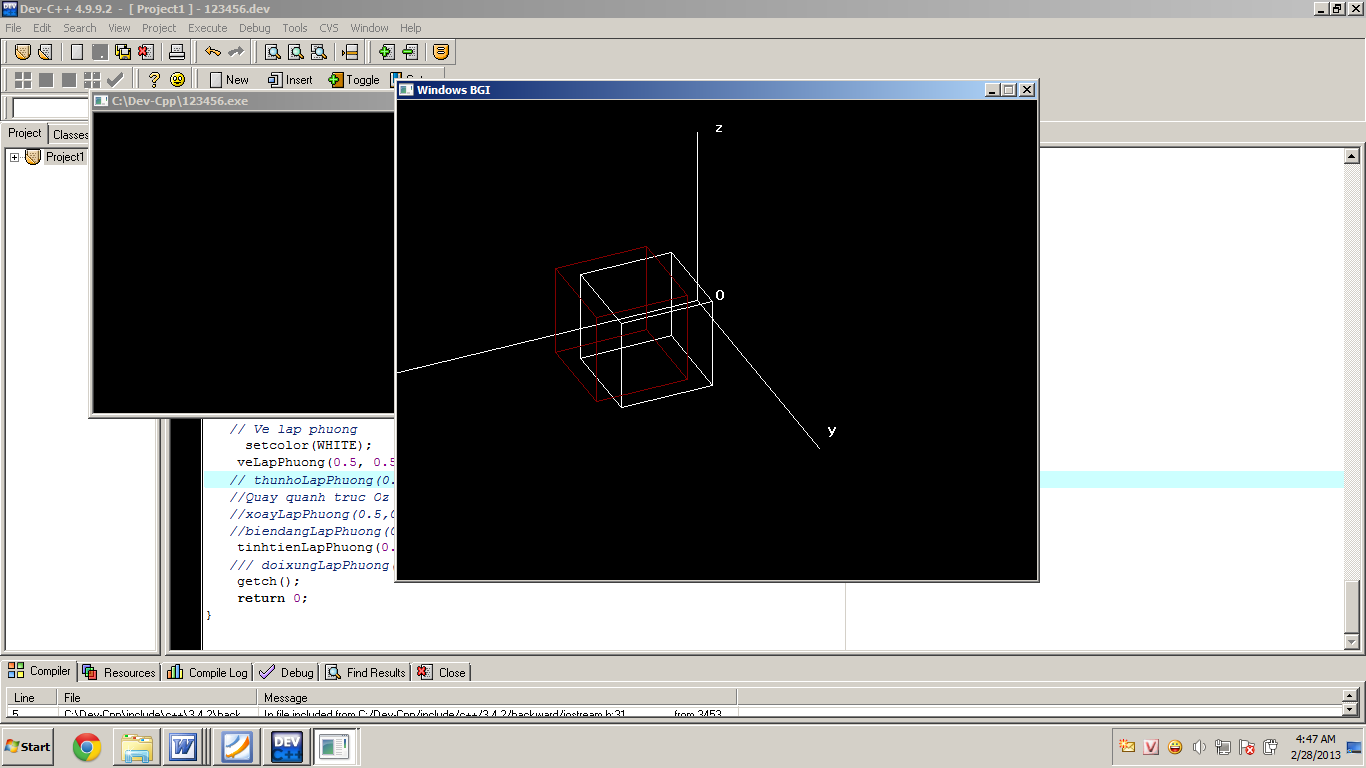
* **Phép quay trục**



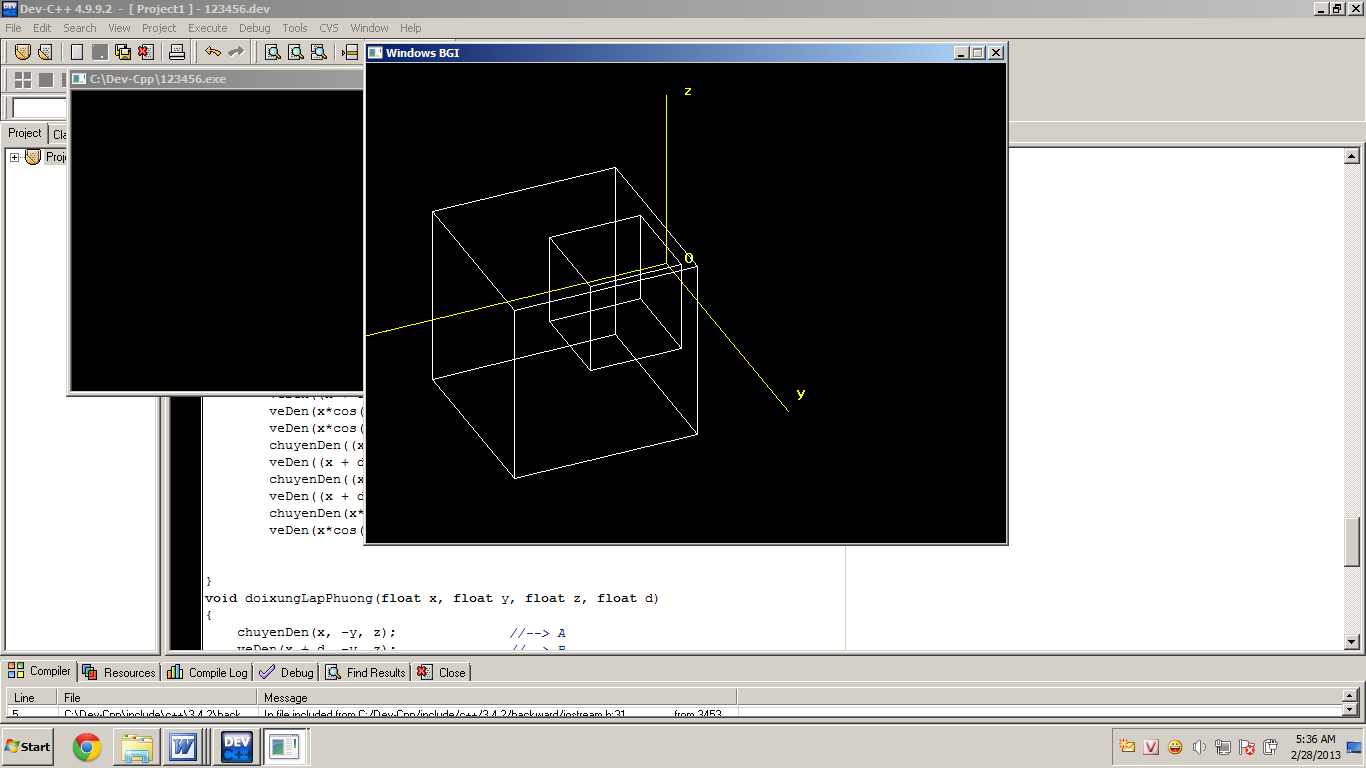
* **Phép phóng to thu nhỏ**



**- Phép tịnh tiến**



- **Phép biến đổi tỷ lệ**



**3.2 Cài đặt chương trình**

#include <dos.h>

#include <math.h>

#include <stdio.h>

#include <conio.h>

#include <iostream.h>

#include <graphics.h>

// Khai bao bien toan cuc

int phepchieu; // = 1 : phoi canh; = 0 : song song

float r = 50, phi = 10,teta = 20,D = 20, tlx = 100, tly = 100;

int xo = 300, yo = 200; // (xo, yo) vi tri cai dat goc toa do tren man hinh

// Xay dung bo cong cu 3D

void chuyenHQS(float x, float y, float z, float &x1, float &y1, float &z1)

{

x1 = - x \* sin(teta) + y \* cos(teta);

y1 = - x \* cos(teta) \* sin(phi) - y \* sin(teta) \* sin(phi) + z \* cos(phi);

z1 = - x \* sin(teta) \* cos(phi) - y \* cos(teta) \* sin(phi) - z \* sin(phi) + r;

}

void chieu3D\_2D(float x, float y, float z, float &xp, float &yp)

{

if (phepchieu == 1)

{

xp = D \* x / z;

yp = D \* y / z;

}

else

{

xp = x;

yp = y;

}

}

void chuyenMH(float x, float y, int &xm, int &ym)

{

xm = (int) (tlx \* x + xo);

ym = (int) (tly \* y + yo);

}

void chuyenDen(float x, float y, float z)

{

float x1, y1, z1, xp, yp;

int xm, ym;

chuyenHQS(x, y, z, x1, y1, z1);

chieu3D\_2D(x1, y1, z1, xp, yp);

chuyenMH(xp, yp, xm, ym);

moveto(xm, ym);

}

void veDen(float x, float y, float z)

{

float x1, y1, z1, xp, yp;

int xm, ym;

chuyenHQS(x, y, z, x1, y1, z1);

chieu3D\_2D(x1, y1, z1, xp, yp);

chuyenMH(xp, yp, xm, ym);

lineto(xm, ym);

}

// Ve truc toa do

void veTruc()

{

// In ky tu

chuyenDen(-0.2, 0, 0.1);

outtext("O");

chuyenDen(4, 0, 0.3);

outtext("x");

chuyenDen(-0.1, 3, 0.3);

outtext("y");

chuyenDen(-0.2, 0, 2.1);

outtext("z");

// Ve truc

// Ox

chuyenDen(0, 0, 0);

veDen(4, 0, 0);

// Oy

chuyenDen(0, 0, 0);

veDen(0, 3, 0);

// Oz

chuyenDen(0, 0, 0);

veDen(0, 0, 2);

}

// Ve lap phuong

void veLapPhuong(float x, float y, float z, float d)

{

chuyenDen(x, y, z); //--> A

veDen(x + d, y, z); //--> B

veDen(x + d, y + d, z); //--> C

veDen(x, y + d, z); //--> D

veDen(x, y, z); //--> A

veDen(x, y, z + d); //--> E

veDen(x + d, y, z + d); //--> F

veDen(x + d, y + d, z + d); //--> G

veDen(x, y + d, z + d); //--> H

veDen(x, y, z + d); //--> E

chuyenDen(x + d, y, z); //--> B

veDen(x + d, y, z + d); //--> F

chuyenDen(x + d, y + d, z); //--> C

veDen(x + d, y + d, z + d); //--> G

chuyenDen(x, y + d, z); //--> D

veDen(x, y + d, z + d); //--> F

}

void biendoitile(float x, float y, float z, float d)

{

chuyenDen(x\*2, y\*2, z\*2); //--> A

veDen((x + d)\*2, y\*2, z); //--> B

veDen((x + d)\*2, (y + d)\*2, z\*2); //--> C

veDen(x\*2, (y + d)\*2, z\*2); //--> D

veDen(x\*2, y\*2, z\*2); //--> A

veDen(x\*2, y\*2, (z + d)\*2); //--> E

veDen((x + d)\*2, y\*2, (z + d)\*2); //--> F

veDen((x + d)\*2, (y + d)\*2, (z + d)\*2); //--> G

veDen(x\*2, (y + d)\*2, (z + d)\*2); //--> H

veDen(x\*2, y\*2, (z + d)\*2); //--> E

chuyenDen((x + d)\*2, y\*2, z\*2); //--> B

veDen((x + d)\*2, y\*2, (z + d)\*2); //--> F

chuyenDen((x + d)\*2, (y + d)\*2, z\*2); //--> C

veDen((x + d)\*2, (y + d)\*2, (z + d)\*2); //--> G

chuyenDen(x\*2, (y + d)\*2, z\*2); //--> D

veDen(x\*2, (y + d)\*2, (z + d)\*2); //--> F

}

void biendangLapPhuong(float x, float y, float z, float d)

{

float a=0.45;

chuyenDen(x+tan(a)\*y, y+x\*tan(a), z); //--> A

veDen((x + d)+y\*tan(a), y+x\*tan(a), z); //--> B

veDen((x + d)+y\*tan(a), (y + d)+x\*tan(a), z); //--> C

veDen(x+y\*tan(a), (y + d)+x\*tan(a), z); //--> D

veDen(x+y\*tan(a), y+x\*tan(a), z); //--> A

veDen(x+y\*tan(a), y+x\*tan(a), z + d); //--> E

veDen((x + d)+y\*tan(a), y+x\*tan(a), z + d); //--> F

veDen((x + d)+y\*tan(a), (y + d)+x\*tan(a), z + d); //--> G

veDen(x+y\*tan(a), (y + d)+x\*tan(a), z + d); //--> H

veDen(x+y\*tan(a), y+x\*tan(a), z + d); //--> E

chuyenDen((x + d)+y\*tan(a), y+x\*tan(a), z); //--> B

veDen((x + d)+y\*tan(a), y+x\*tan(a), z + d); //--> F

chuyenDen((x + d)+y\*tan(a), (y + d)+x\*tan(a), z); //--> C

veDen((x + d)+y\*tan(a), (y + d)+x\*tan(a), z + d); //--> G

chuyenDen(x+y\*tan(a), (y + d)+x\*tan(a), z); //--> D

veDen(x+y\*tan(a), (y + d)+x\*tan(a), z + d); //--> F

}

void tinhtienLapPhuong(float x,float y,float z,float d)

{

setcolor(BLACK);

setcolor(WHITE);

veTruc();

setcolor(4);

chuyenDen(x+0.5, y+0.5, z+0.5); //--> A

veDen(x + d+0.5, y+0.5, z+0.5); //--> B

veDen(x + d+0.5, y + d+0.5, z+0.5); //--> C

veDen(x+0.5, y + d+0.5, z+0.5); //--> D

veDen(x+0.5, y+0.5, z+0.5); //--> A

veDen(x+0.5, y+0.5, z + d+0.5); //--> E

veDen(x + d+0.5, y+0.5, z + d+0.5); //--> F

veDen(x + d+0.5, y + d+0.5, z + d+0.5); //--> G

veDen(x+0.5, y + d+0.5, z + d+0.5); //--> H

veDen(x+0.5, y+0.5, z + d+0.5); //--> E

chuyenDen(x + d+0.5, y+0.5, z+0.5); //--> B

veDen(x + d+0.5, y+0.5, z + d+0.5); //--> F

chuyenDen(x + d+0.5, y + d+0.5, z+0.5); //--> C

veDen(x + d+0.5, y + d+0.5, z + d+0.5); //--> G

chuyenDen(x+0.5, y + d+0.5, z+0.5); //--> D

veDen(x+0.5, y + d+0.5, z + d+0.5);

}

void thunhoLapPhuong(float x,float y,float z,float d)

{

setcolor(BLACK);

setcolor(WHITE);

veTruc();

setcolor(5);

chuyenDen(x\*0.5, y\*0.5, z\*0.5); //--> A

veDen((x + d)\*0.5, y\*0.5, z\*0.5); //--> B

veDen((x + d)\*0.5, (y + d)\*0.5, z\*0.5); //--> C

veDen(x\*0.5, (y + d)\*0.5, z\*0.5); //--> D

veDen(x\*0.5, y\*0.5, z\*0.5); //--> A

veDen(x\*0.5, y\*0.5, (z + d)\*0.5); //--> E

veDen((x + d)\*0.5, y\*0.5, (z + d)\*0.5); //--> F

veDen((x + d)\*0.5,(y + d)\*0.5, (z + d)\*0.5); //--> G

veDen(x\*0.5, (y + d)\*0.5, (z + d)\*0.5); //--> H

veDen(x\*0.5, y\*0.5, (z + d)\*0.5); //--> E

chuyenDen((x + d)\*0.5, y\*0.5, z\*0.5); //--> B

veDen((x + d)\*0.5, y\*0.5, (z + d)\*0.5); //--> F

chuyenDen((x + d)\*0.5, (y + d)\*0.5, z\*0.5); //--> C

veDen((x + d)\*0.5, (y + d)\*0.5, (z + d)\*0.5); //--> G

chuyenDen(x\*0.5, (y + d)\*0.5, z\*0.5); //--> D

veDen(x\*0.5, (y + d)\*0.5, (z + d)\*0.5);

}

void xoayLapPhuong(float x,float y,float z,float d){

setcolor(BLACK);

float goc;

cout<<"nhap goc quay";cin>>goc;

float a=(goc\*3.14152)/180;

setcolor(WHITE);

veTruc();

setcolor(YELLOW);

chuyenDen(x\*cos(a) - y\*sin(a), x\*sin(a) + y\*cos(a), z); //--> A

veDen((x + d)\*cos(a) - y\*sin(a),(x + d)\*sin(a) + y\*cos(a), z); //--> B

veDen((x + d)\*cos(a) - (y+d)\*sin(a) , (x + d)\*sin(a) + (y+d)\*cos(a), z); //--> C

veDen(x\*cos(a) - (y + d)\*sin(a),x\*sin(a) + (y+ d)\*cos(a), z); //--> D

veDen(x\*cos(a) - y\*sin(a), x\*sin(a) + y\*cos(a), z); //--> A

veDen(x\*cos(a) - y\*sin(a), x\*sin(a) + y\*cos(a), z + d); //--> E

veDen((x + d)\*cos(a) - y\*sin(a),(x + d)\*sin(a) + y\*cos(a), z + d); //--> F

veDen((x + d)\*cos(a) - (y+d)\*sin(a) , (x + d)\*sin(a) + (y+d)\*cos(a), z + d); //--> G

veDen(x\*cos(a) - (y + d)\*sin(a),x\*sin(a) + (y+ d)\*cos(a), z + d); //--> H

veDen(x\*cos(a) - y\*sin(a), x\*sin(a) + y\*cos(a), z + d); //--> E

chuyenDen((x + d)\*cos(a) - y\*sin(a),(x + d)\*sin(a) + y\*cos(a), z); //--> B

veDen((x + d)\*cos(a) - y\*sin(a),(x + d)\*sin(a) + y\*cos(a), z + d); //--> F

chuyenDen((x + d)\*cos(a) - (y+d)\*sin(a) , (x + d)\*sin(a) + (y+d)\*cos(a), z); //--> C

veDen((x + d)\*cos(a) - (y+d)\*sin(a) , (x + d)\*sin(a) + (y+d)\*cos(a), z + d); //--> G

chuyenDen(x\*cos(a) - (y + d)\*sin(a),x\*sin(a) + (y+ d)\*cos(a), z); //--> D

veDen(x\*cos(a) - (y + d)\*sin(a),x\*sin(a) + (y+ d)\*cos(a), z + d); //--> F

}

void doixungLapPhuong(float x, float y, float z, float d)

{

chuyenDen(x, -y, z); //--> A

veDen(x + d, -y, z); //--> B

veDen(x + d, -(y + d),z); //--> C

veDen(x, -(y + d), z); //--> D

veDen(x, -y, z); //--> A

veDen(x, -y, (z + d)); //--> E

veDen(x + d, -y, (z + d)); //--> F

veDen(x + d, -(y + d), (z + d)); //--> G

veDen(x,-(y + d), (z + d)); //--> H

veDen(x, -y, (z + d)); //--> E

chuyenDen(x + d, -y, z); //--> B

veDen(x + d, -y, (z + d)); //--> F

chuyenDen(x + d, -(y + d), z); //--> C

veDen(x + d, -(y + d), (z + d)); //--> G

chuyenDen(x, -(y + d), z); //--> D

veDen(x, -(y + d), (z + d)); //--> F

}

int main()

{

int driver = DETECT, mode = 0;

initgraph(&driver, &mode,"C:\\Dev-Cpp\\include");

// Ve truc toa do

setcolor(YELLOW);

veTruc();

// Ve lap phuong

setcolor(WHITE);

veLapPhuong(0.5, 0.5, 0, 1);

// thunhoLapPhuong(0.5,0.5,0,1);

//Quay quanh truc Oz

biendoitile(0.5,0.5,0,1);

//xoayLapPhuong(0.5,0.5,0,1);

//biendangLapPhuong(0.5,0.5,0,1);//sai

//tinhtienLapPhuong(0.5,0.5,0,1);

/// doixungLapPhuong(0.5,0.5,0,1);

getch();

return 0;

}

**TỰ ĐÁNH GIÁ**

*Ưu điểm* : - Thiết thực dễ dàng sử dụng và nâng cấp

- Chiếm ít tài nguyên của máy.

-Tiết kiệm điện khi máy không được sử dụng.

*Khuyết điểm* : Còn thiếu 1 số chức năng.

*Hướng phát triển* : Trao đổi thông tin trên mạng.

KẾT LUẬN

Qua quá trình tìm hiểu và phân tích đề tài “Viết chương trình mô phỏng phép tịnh tiến, phép biến đổi tỉ lệ, phép quay trục, phóng to thu nhỏ, biến dạng của một hình bất kỳ trong không gian 3 chiều”. Chúng em đã hiểu được cách phân tích và thiết kế một phần mềm. Tuy nhiên do kiến thức còn hạn hẹp nên chúng em chưa thể phân tích được sâu sắc hơn và đưa ra được một phần mềm hoàn chỉnh, nhưng em mong rằng qua đề tài này các bạn có thể hiểu thêm nhiều về các phép vẽ cơ bản trong môn học đồ họa máy tính.