

**SUMMER PROJECT 2011**

**PROGRAMMING CLUB**

**IIT KANPUR**

**CRYSTOSIM**

<b>NAME:</b>	ASHITA PRASAD
<b>ROLL:</b>	10159
<b>DEPARTMENT:</b>	MATERIAL SCIENCE AND ENGINEERING
<b>e-mail:</b>	ashita@iitk.ac.in
<b>MENTOR:</b>	ANKIT MAHATO

# ACKNOWLEDGEMENT:

I would like to express my sincere gratitude to all those people who have been associated with this project and have helped me with it and made it a worthwhile experience.

I would specially like to thank the coordinators of the programming club and my mentor Ankit Mahato. I would also like to thank my parents for their constant support. Without their help and guidance this project would not have been materialized.

Ashita Prasad

# PROLOGUE:

## PROBLEM STATEMENT:

To develop platform independent simulation software which is capable of simulating 3D crystal structures, crystallographic planes and directions for all lattices including hexagonal lattice.

## USP OF THE PROJECT:

Last year seniors in the club had explored vpython. But this year I tried to integrate vpython with the gui in a frame. Also this is perhaps the first open source software demonstrating crystallographic planes and directions for all crystal systems including the hexagonal lattice.

## PLATFORM USED

I have used python(version python 2.7) as my programming language for the development of this software. For the generation of simulated figures I have used Vpython. I have integrated vpython with the GUI in a frame using wxpython.



# Basic Crystal Concepts

## Unit Cell

A crystal is an array of atoms packed together in a regular pattern. A *unit cell* of a pattern is a piece of the pattern which, when repeated through space without rotation and without gaps or overlaps, reconstructs the pattern to infinity. For filling space without holes, a unit cell must be either a parallelogram (in 2D) or a parallelepiped (in 3D).

The symmetries of a pattern determine the shape of the unit cell. For example, mirror symmetry requires a rectangular (in 2D) or tetragonal (in 3D) unit cell. There is an infinite number of possible unit cells for any pattern (*e. g.*, a given unit cell can generate a family of unit cells by repeated doublings in size). By custom, the unit cell is chosen to be the smallest one that reveals the special geometry characteristic of the symmetry. Thus, although an oblique parallelogram can be used for a pattern with 4-fold symmetry in 2D, a square is preferred.

When a unit cell is repeatedly translated to fill all of 2D or 3D space, the vertices of all the unit cells in the filled space constitute a *lattice*. A lattice is an infinite array of regularly-spaced points. All points in the lattice have identical "environments" --- the view from every point in the lattice is identical to that from any other point in the lattice. The absolute positions of the points of a lattice, and hence the unit cell, are arbitrary with respect to a pattern.

Not all lattice points need coincide with unit cell vertices. *Primitive* unit cells use every lattice point as a unit cell vertex. *Non-primitive* unit cells, however, contain extra lattice points not at the corners.

A primitive unit cell contains exactly one lattice point. For example in 2D, each primitive unit cell joins four lattice points, each of which counts for 1/4 because every lattice point is shared among four unit cells. In 2D, a non-primitive unit cell has one additional lattice point exactly centered within it and is called a *body-centered* non-primitive unit cell. In 3D, non-primitive cells are of three kinds:

- *end-centered* : an extra lattice point is centered in each of two opposing faces of the cell
- *face-centered* : an extra lattice point is centered in every face of the cell
- *body-centered* : an extra lattice point is centered in the exact middle of the cell

Although primitive unit cells are simpler than non-primitive unit cells, the non-primitive unit cell is preferred when its geometry is more favorable (simpler). For instance, a rectangular non-primitive cell would be chosen over a rhomboid primitive cell. In general, the unit cell used is the smallest one with the most regular geometry.

# Lattice and Unit Cell Parameters

A lattice may be specified by two non-coincident vectors in 2D, and by three non-coplanar vectors in 3D. The vectors lie along the edges of the unit cell, and are labeled  $a$ ,  $b$ , and (in 3D)  $c$ . The magnitude of the vectors is given by the dimensions of the unit cell in the real crystal under study.

The faces of the unit cell are labeled as follows:

- $A$  : edges defined by lattice vectors  $b$  and  $c$
- $B$  : edges defined by lattice vectors  $a$  and  $c$
- $C$  : edges defined by lattice vectors  $a$  and  $b$

Similarly, the *inter-facial angles* of the unit cell are defined to be:

- $\alpha$  : angle between edges  $b$  and  $c$
- $\beta$  : angle between edges  $a$  and  $c$
- $\gamma$  : angle between edges  $a$  and  $b$

## Lattice Systems: the 14 Bravais Lattices

Lattices can be classified into "systems", each system being characterized by the shape of its associated unit cell. In three dimensions, the lattices are categorized into seven crystal lattice "systems". Within several of these, lattices supporting non-primitive unit cells can be defined. The classification scheme yields a total of 14 possible lattices (called *Bravais* lattices).

The lattice symbols used for classification are as follows:

- P : primitive
- B : end-centered on B-face (convention for Monoclinic systems)
- C : end-centered on C-face (convention for Orthorhombic systems)
- I : body-centered
- F : face-centered
- R : rhombohedral primitive

System	# of lattices in system	Lattice symbols
Triclinic (Anorthic)	1	P
Monoclinic	2	P, B
Orthorhombic	4	P, C, I, F

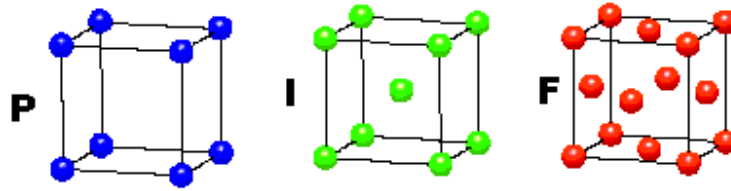
Tetragonal	2	P, I
Isometric (Cubic)	3	P, I, F
Trigonal/Rhombohedral	1	P or R
Hexagonal	1	P

In two dimensions, there are only four possible unit cell shapes and two possible lattice symbols:

<b>Cell shape</b>	<b>Lattice symbol</b>
General parallelogram (rhomboid)	P
Rectangle	p, c
Square	P
Rhombus with 60 degree angle	P

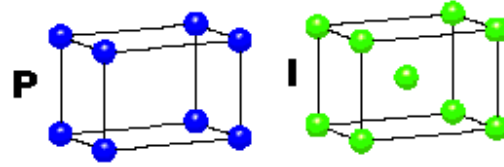
### CUBIC

$$a = b = c$$
$$\alpha = \beta = \gamma = 90^\circ$$



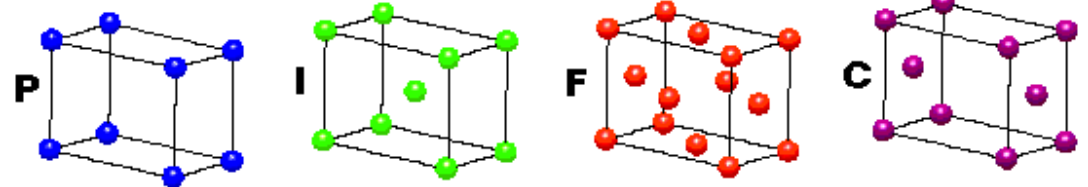
### TETRAGONAL

$$a = b \neq c$$
$$\alpha = \beta = \gamma = 90^\circ$$



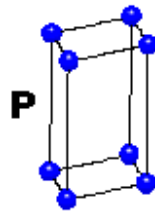
### ORTHORHOMBIC

$$a \neq b \neq c$$
$$\alpha = \beta = \gamma = 90^\circ$$



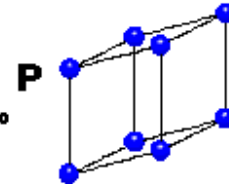
### HEXAGONAL

$$a = b \neq c$$
$$\alpha = \beta = 90^\circ$$
$$\gamma = 120^\circ$$



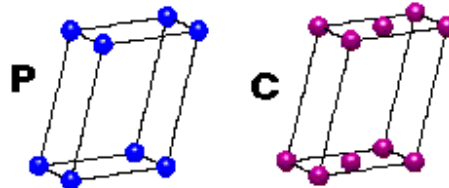
### TRIGONAL

$$a = b = c$$
$$\alpha = \beta = \gamma \neq 90^\circ$$



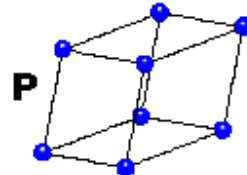
### MONOCLINIC

$$a \neq b \neq c$$
$$\alpha = \gamma = 90^\circ$$
$$\beta \neq 120^\circ$$



### TRICLINIC

$$a \neq b \neq c$$
$$\alpha \neq \beta \neq \gamma \neq 90^\circ$$



#### 4 Types of Unit Cell

P = Primitive

I = Body-Centred

F = Face-Centred

C = Side-Centred

+

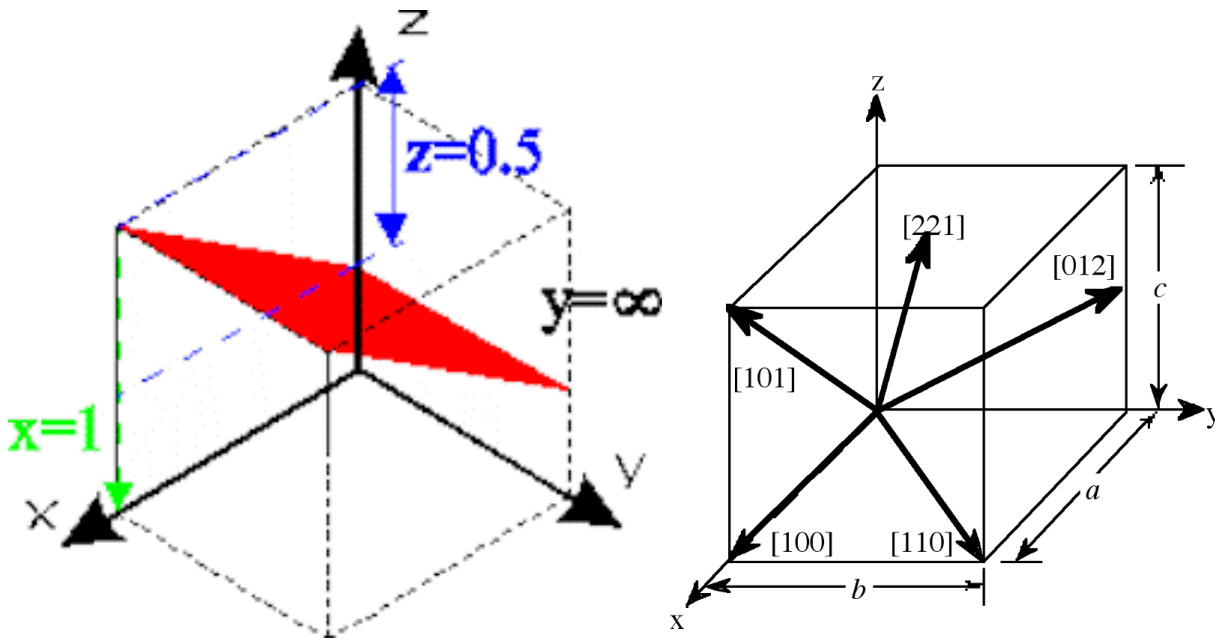
7 Crystal Classes

→ 14 Bravais Lattices

# CRYSTALLOGRAPHIC PLANES AND DIRECTIONS

## Miller Indices

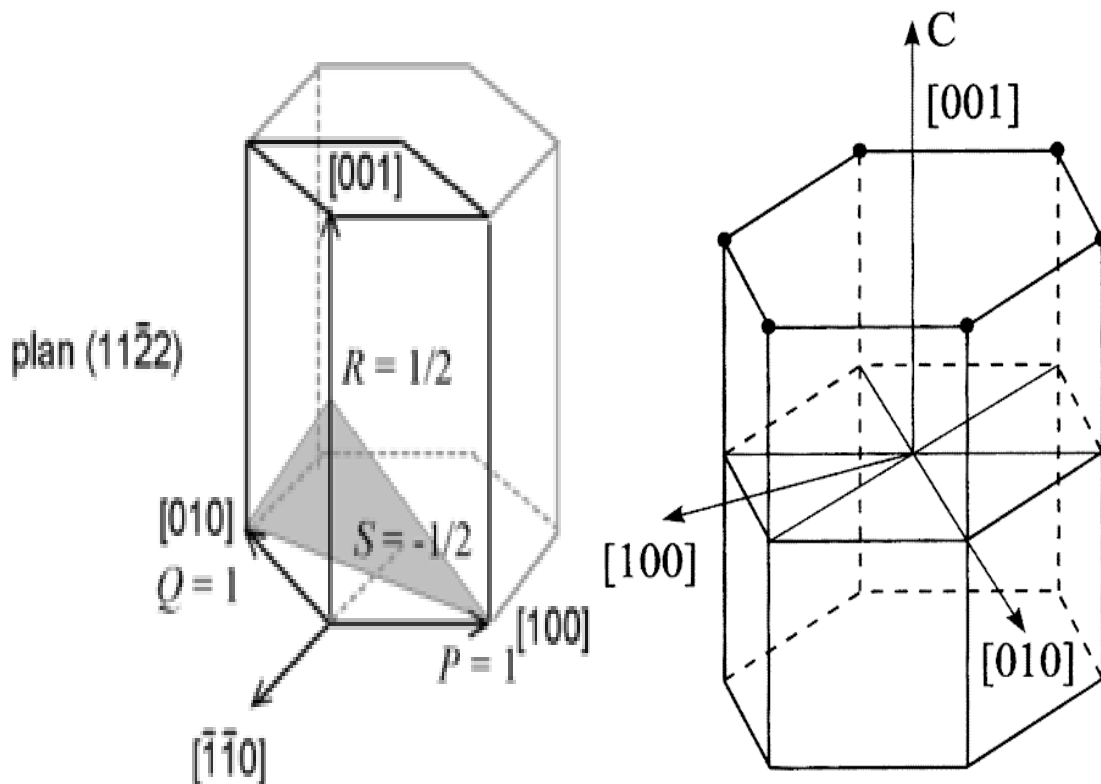
- The orientation of a surface or a crystal plane may be defined by considering how the plane (or indeed any parallel plane) intersects the main crystallographic axes of the solid. The application of a set of rules leads to the assignment of the Miller Indices,  $(hkl)$ ; a set of numbers which quantify the intercepts and thus may be used to uniquely identify the plane or surface.
- A set of parallel crystallographic planes is indicated by its Miller Index  $(hkl)$ . The Miller Index of a plane is derived from the intercepts of the plane with the crystallographic axes



## In case of hexagonal system

### HEXAGONAL CRYSTALS

- For crystals having hexagonal symmetry, it is desirable that equivalent planes have the same indices; as with directions, this is accomplished by the Miller–Bravais system shown in Figure 3.21.
- This convention leads to the four-index (hkil) scheme, which is favored in most instances, since it more clearly identifies the orientation of a plane in a hexagonal crystal.
- There is some redundancy in that  $i$  is determined by the sum of  $h$  and  $k$  through  $i = -(h + k)$
- Otherwise the three  $h$ ,  $k$ , and  $l$  indices are identical for both indexing systems. Figure 3.22b presents several of the common planes that are found for crystals having hexagonal symmetry.



# ALGORITHM FOR THE GENERATION OF

## BRAVAIS LATTICES

For the generation of lattices where the angle between the faces are 90, the end points of the lattices were determined using for loop and simultaneously lines were constructed in x,y and z directions as required to make the lattice structure. The spheres with radius equal to half the smallest side were also placed at correct positions depending on the type of subdivision.

For the lattice subdivisions:

### Primitive:

In this case spheres were simply placed at the lattice points generated by the loop.

### Body Centered:

In this case in addition to the end points the spheres were also placed at the centre of each unit cell.

Starting from the midpoint of each unit cell ( $a/2, b/2$  and  $c/2$ ), by incrementing it with  $a, b$  and  $c$  as required all the midpoints of the unit cells were traversed and the spheres were placed there.

### End Centered:

In this case in addition to the end points of the lattice the spheres were also placed at the centre of two opposite faces. Centre of the required pair of faces were determined and the spheres were placed.

### Face Centered:

In this case in addition to the end points of the lattice the spheres were also placed at the centre all the faces.

Centers of each pair of opposite faces were determined using loop and the atoms were placed there.

# CASE OF OTHER CRYSTAL SYSTEMS

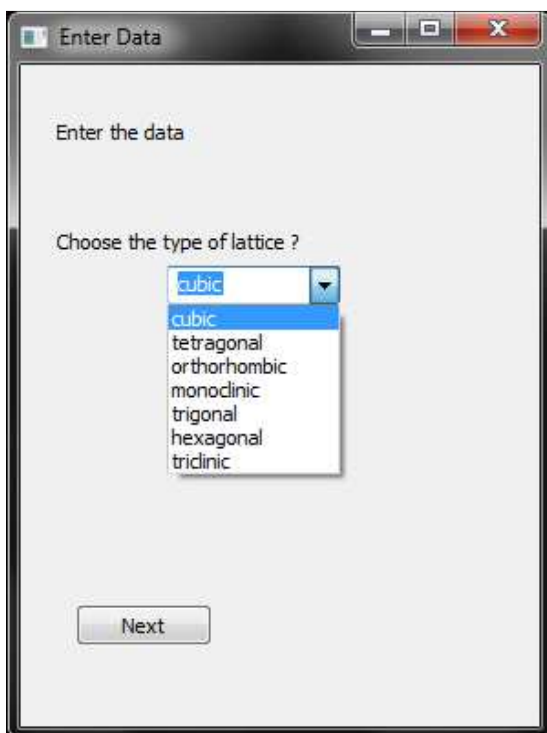
Coming to the case of the lattices where the angles were not inclined at 90 and had arbitrary angles, the vector method was employed to create the lattice structure.

Firstly one line was assumed as the reference or the x axis. From that line respectively lines were constructed at the appropriate angles using the vector algebra and then using vector addition the whole lattice structure was developed. When the whole large lattice structure was created it was appropriately divided into required no of unit cells depending on the value of l, w and h.

## CUBIC

In the cubic module, the parameters taken are

- a which denotes the length of each side of the cube
- l which denotes the no of cubes present along the x axis ie length
- w which denotes the no of cubes present along y axis ie width
- h which denotes the no of cubes present along the z axis ie depth
- p which denotes the percentage in which each sphere is to be displayed



## PRIMITIVE :

Enter Data

Enter the values

Choose the subdivision

primitive

a 1.6

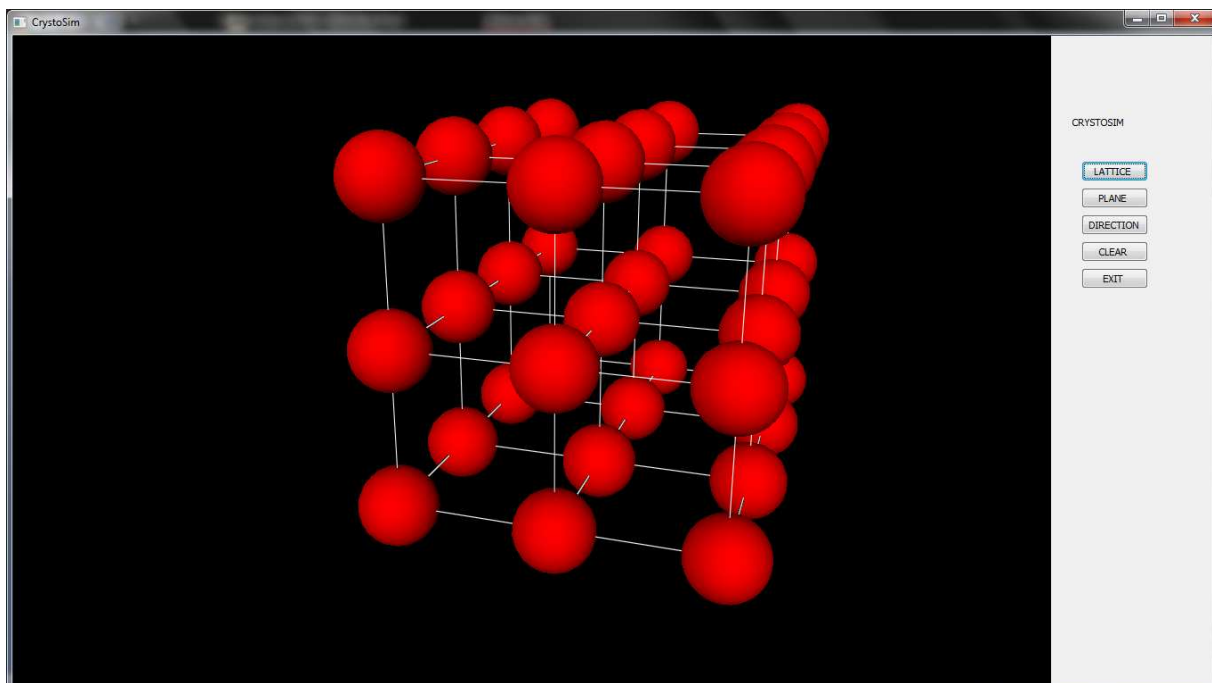
percentage : 50

l 2

w 2

h 3

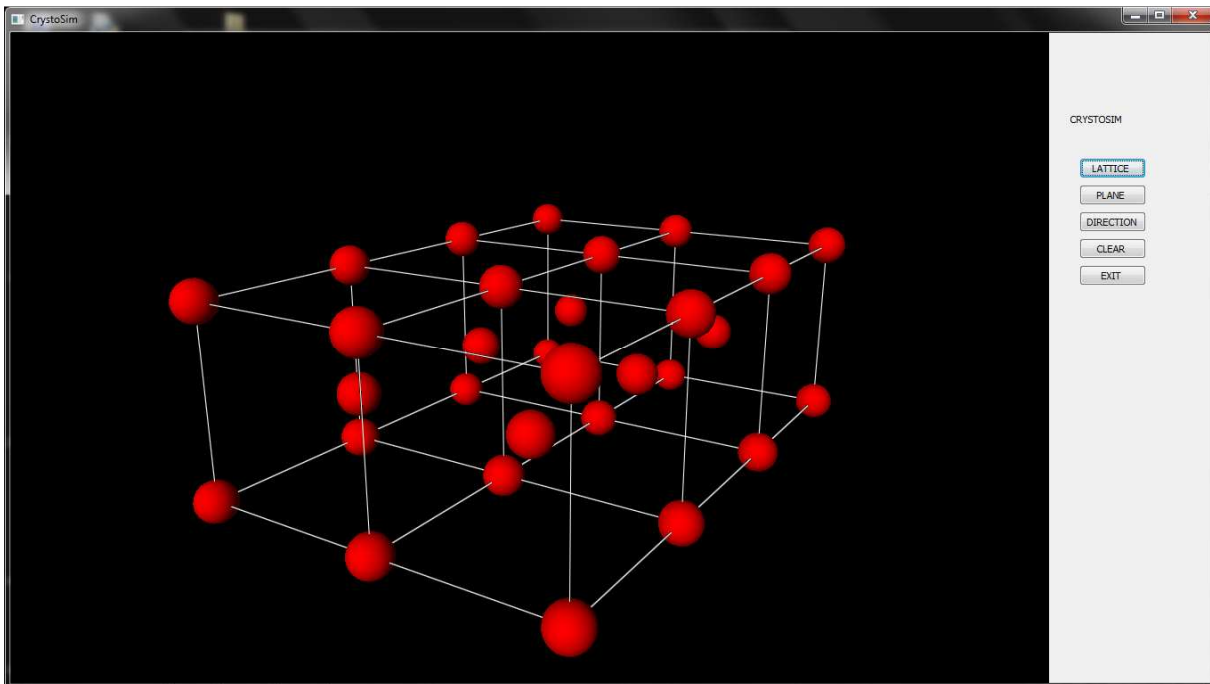
OK



## BODY CENTERED:

The values of the parameters entered are:

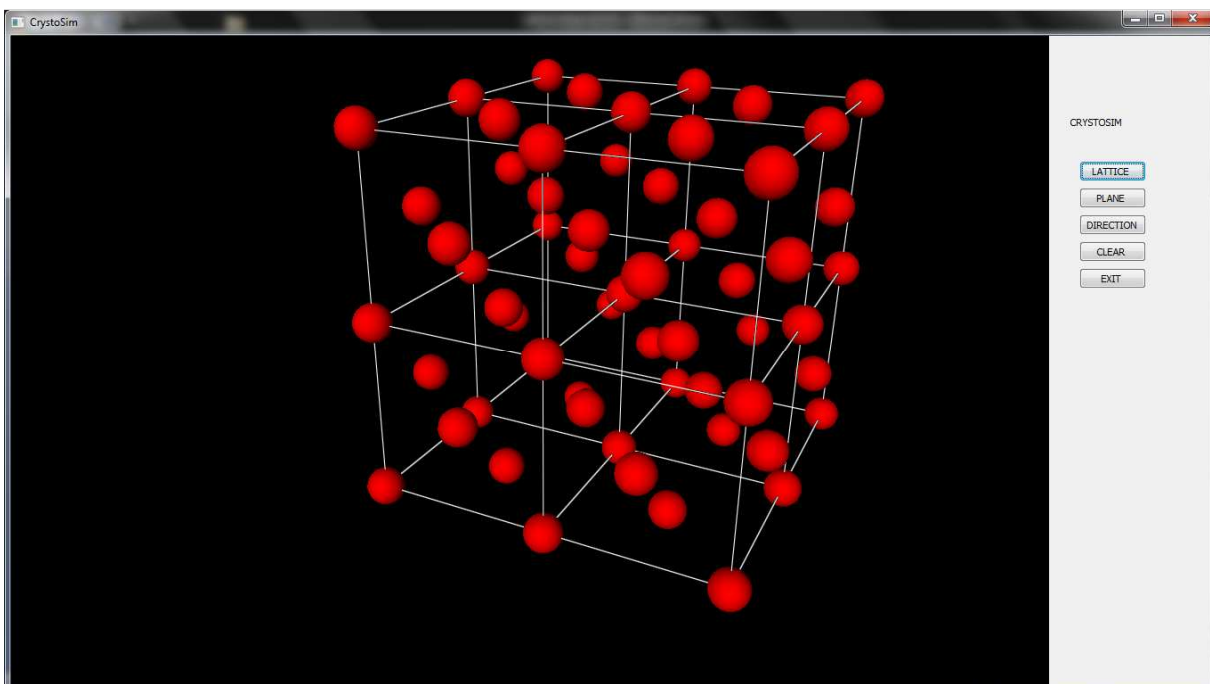
$a=2.2$ ,  $p=25$ ,  $l=2$ ,  $w=1$ ,  $h=3$



## FACE CENTERED:

Values of the parameters for the figure displayed below

$a=1$  ,  $p=30$  ,  $l=2$  ,  $w=2$  ,  $h=2$



## TETRAGONAL:

Primitive and bodycentered lattices are made in the tetragonal and tetragonalbc module respectively

The parameters taken are :

a,c      which denotes the length of different sides of the crystal structure

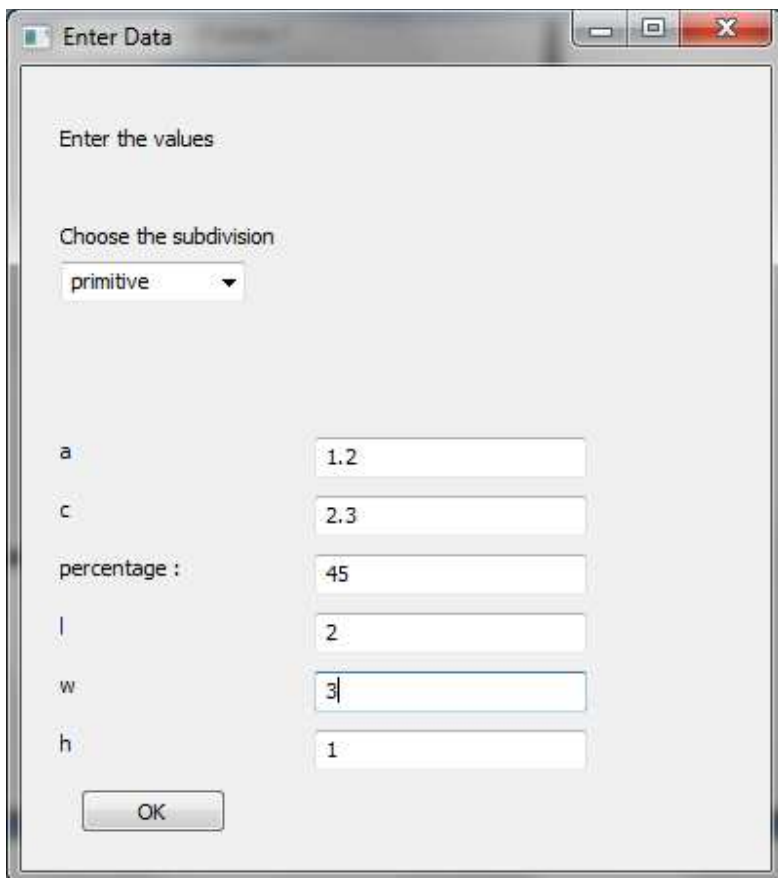
l         which denotes the no of unit cells present along the x axis ie length

w         which denotes the no of unit cells present along y axis ie width

h         which denotes the no of unit cells present along the z axis ie depth

p         which denotes the percentage in which each sphere is to be displayed

## PRIMITIVE:



Enter Data

Enter the values

Choose the subdivision

primitive

a      1.2

c      2.3

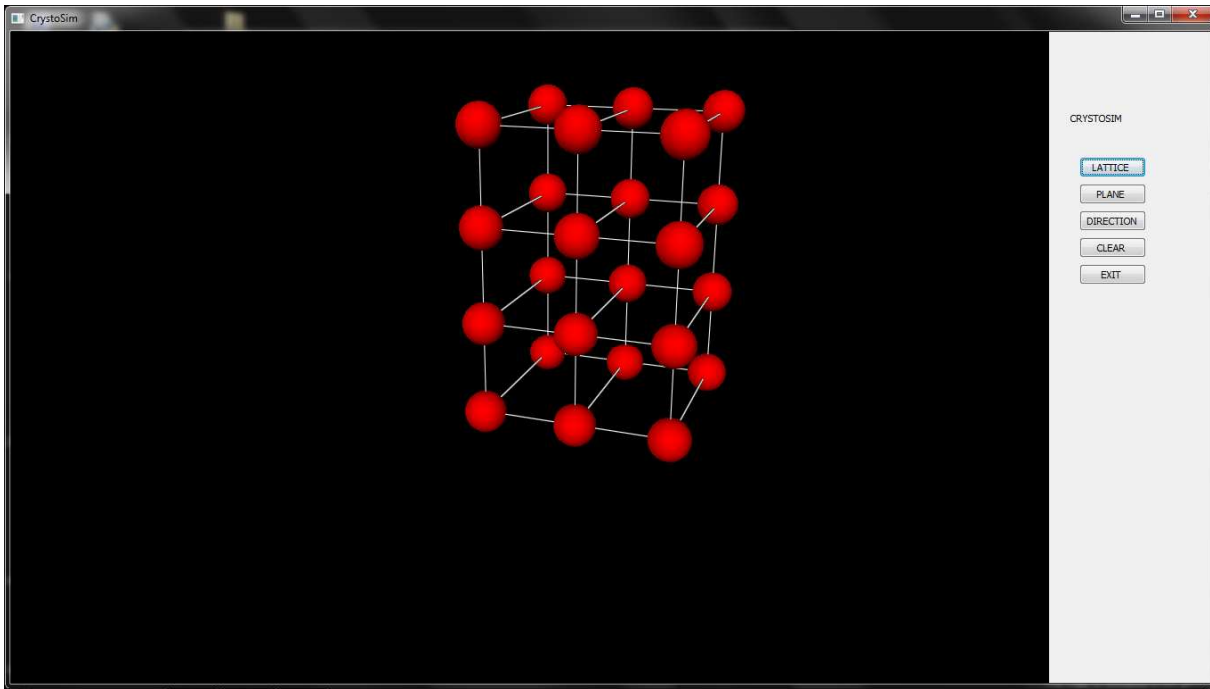
percentage :      45

l      2

w      3

h      1

OK

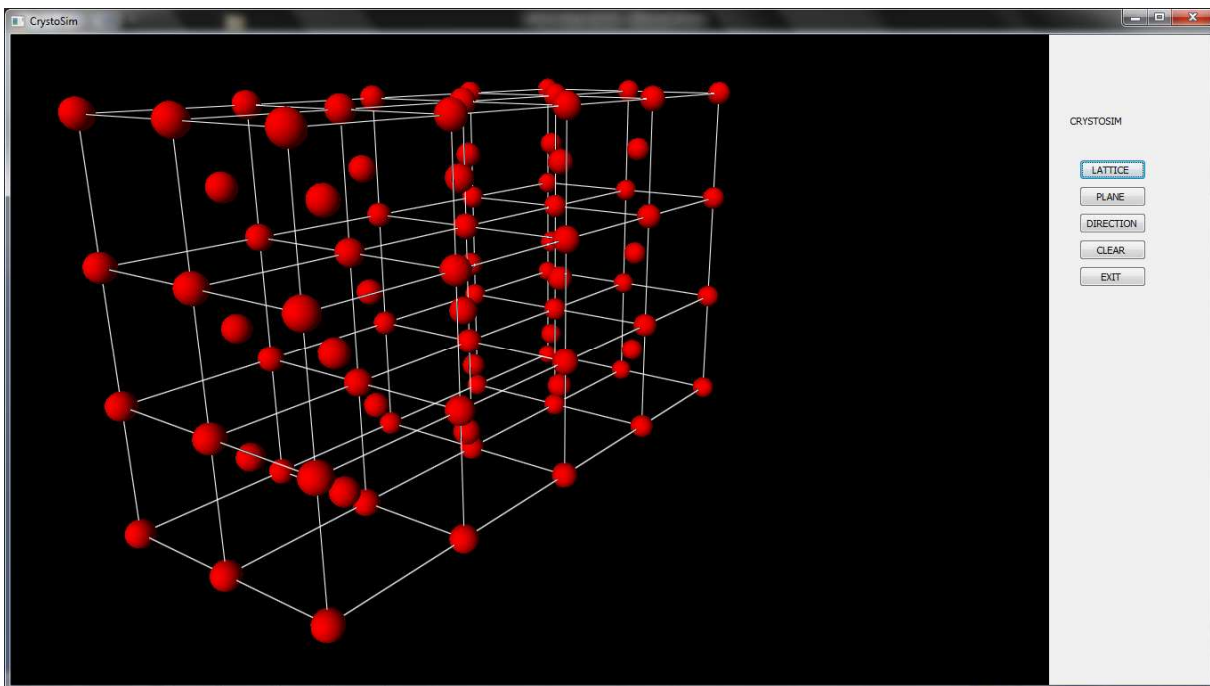


### **BODY CENTERED:**

An example is shown below.

Parameters entered were:

$a=1.3$  ,  $c=3.4$  ,  $p=25$  ,  $l=2$  ,  $w=3$  ,  $h=4$



## ORTHORHOMBIC

The parameters taken are :

a,b,c denotes the length of different sides of the crystal structure in each of the three directions x , y and z

l denotes the no of unit cells present along the x axis ie length

w denotes the no of unit cells present along y axis ie width

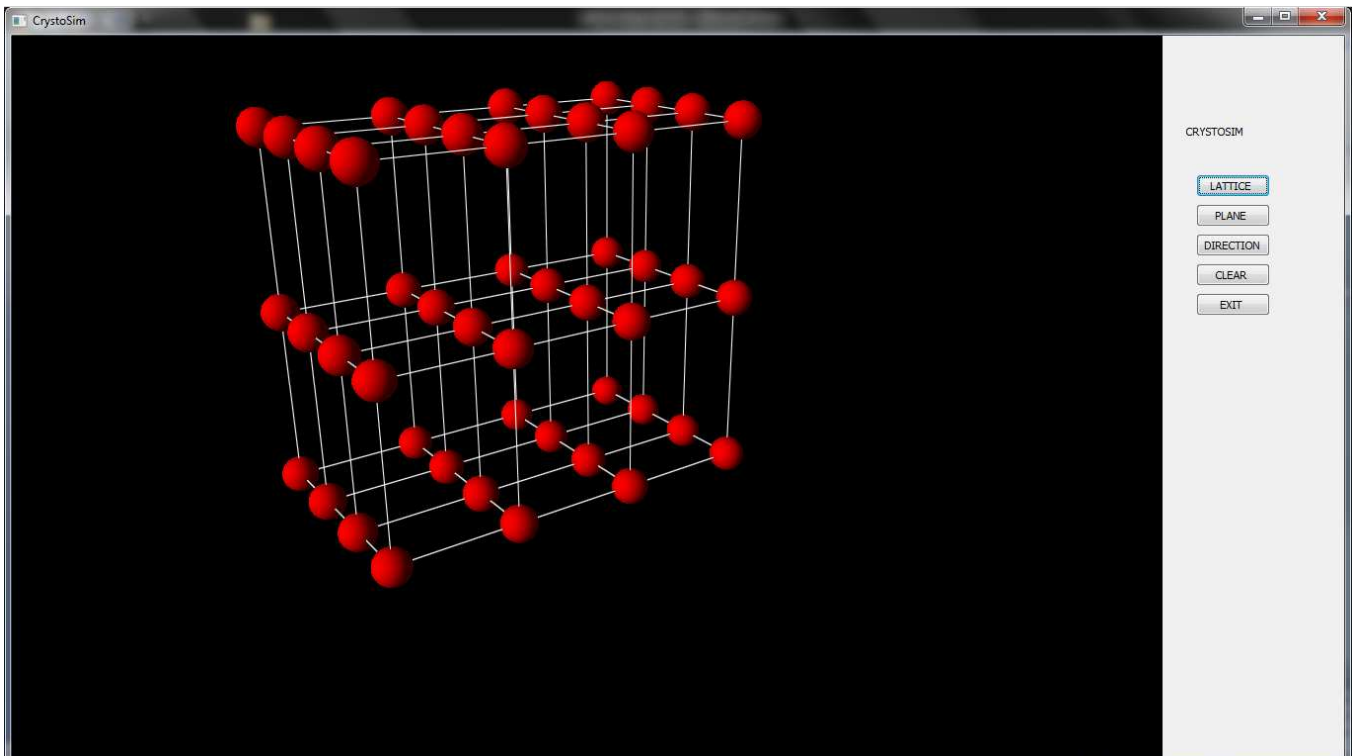
h denotes the no of unit cells present along the z axis ie depth

p denotes the percentage in which each sphere is to be displayed

## PRIMITIVE

The parameters entered are:

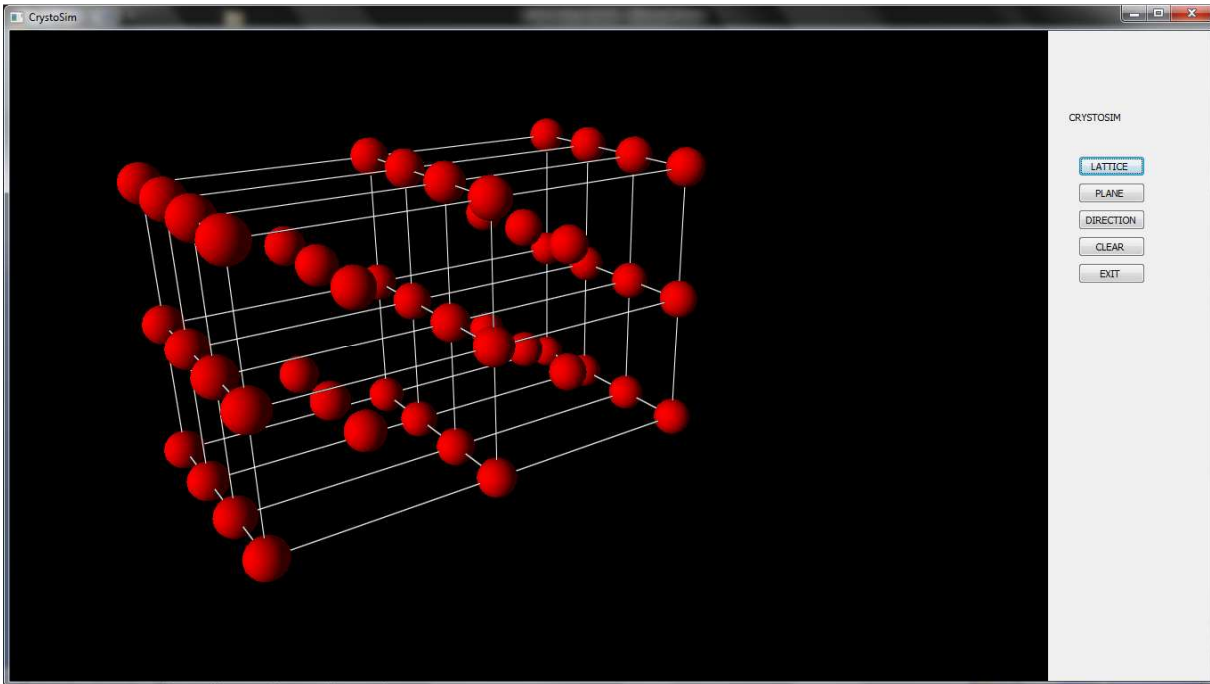
a=1.3 , b=3.1 , c=2.3 ,p=50 , l=3 ,w=2 ,h=3



## BODY CENTERED:

Parameters entered for the example shown below :

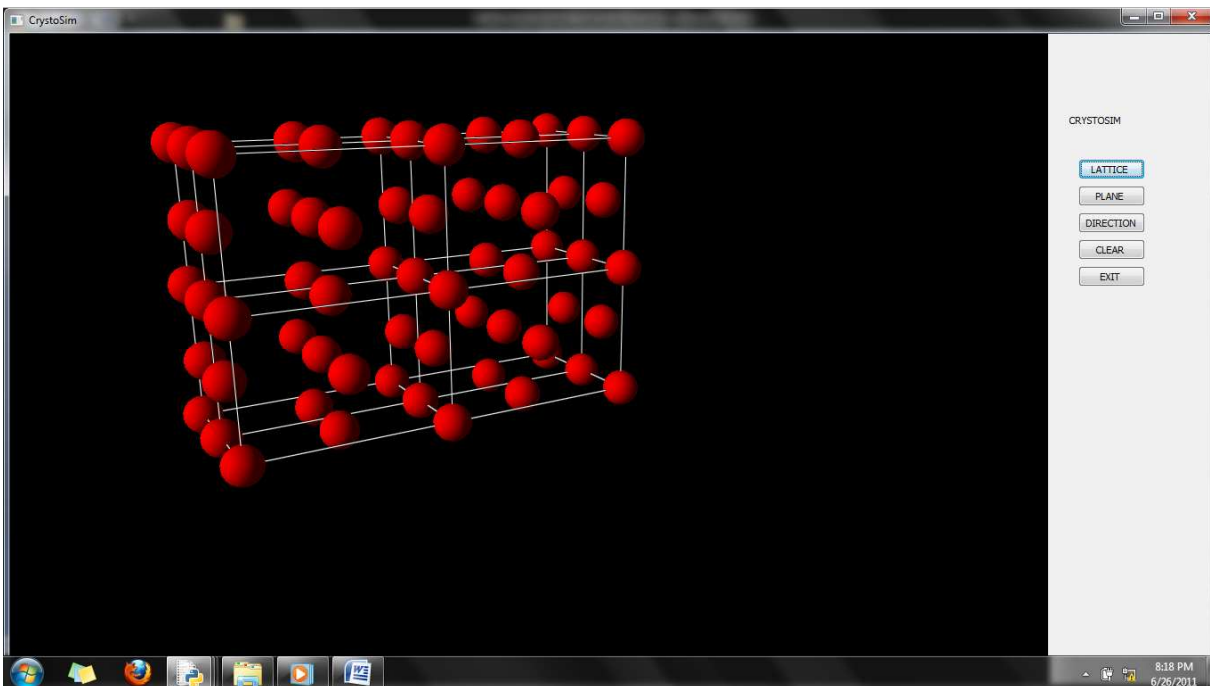
$a=1.2$  ,  $b=2.1$  ,  $c=3.5$  ,  $p=50$  ,  $l=3$  ,  $w=2$  ,  $h=2$



## FACE CENTERED:

Parameters entered for the example shown below:

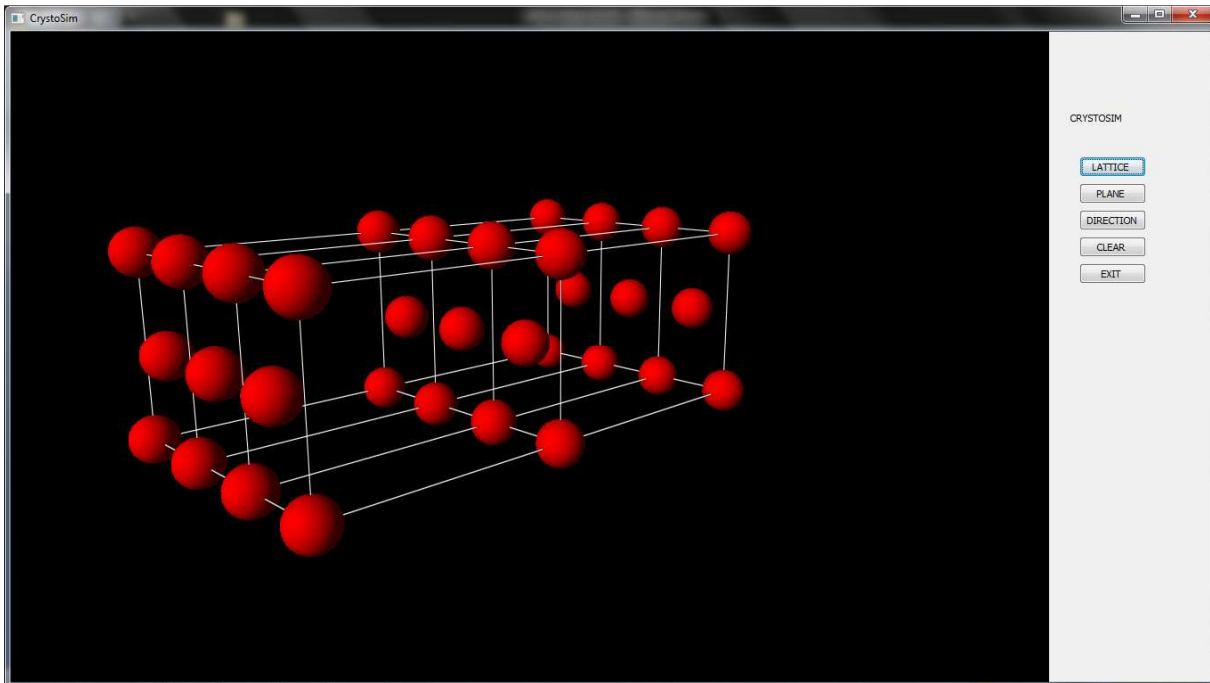
$a=1.2$  ,  $b=2.1$  ,  $c=3.2$  ,  $p=50$  ,  $l=2$  ,  $w=2$  ,  $h=2$



## END CENTERED:

Parameters entered for the example shown below:

$a=1.1$  ,  $b=2.1$  ,  $c=3.1$  ,  $p=50$  ,  $l=3$  ,  $w=1$  ,  $h=2$



## OTHER CRYSTAL SYSTEMS:

In the case of other crystal systems where the angle between the faces are not 90 , vector method has been used .

The parameters used in the program are:

- |                              |  |
|------------------------------|--|
| $a,b,c$                      | which denotes the length of different sides of the crystal structure in each of the three directions $x$ $y$ and $z$ |
| $ma$                         | the no of unit cells present along the $x$ axis ie length  |
| $mb$                         | the no of unit cells present along $y$ axis ie width   |
| $mc$                         | the no of unit cells present along the $z$ axis ie depth   |
| $p$                          | the percentage in which each sphere is to be displayed   |
| $\alpha, \beta$ and $\gamma$ | : angles between the $xy$ , $yz$ and $zx$ plane respectively   |

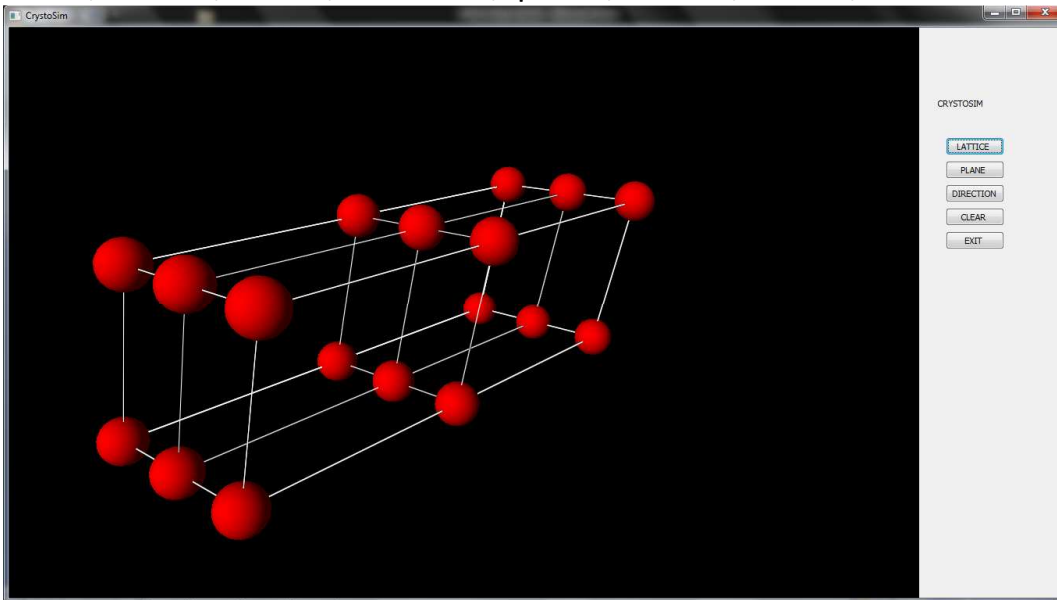
## MONOCLINIC:

In this case alpha and gamma are fixed at 90. Only the beta angle varies.

### PRIMITIVE:

The values of the parameters entered are :

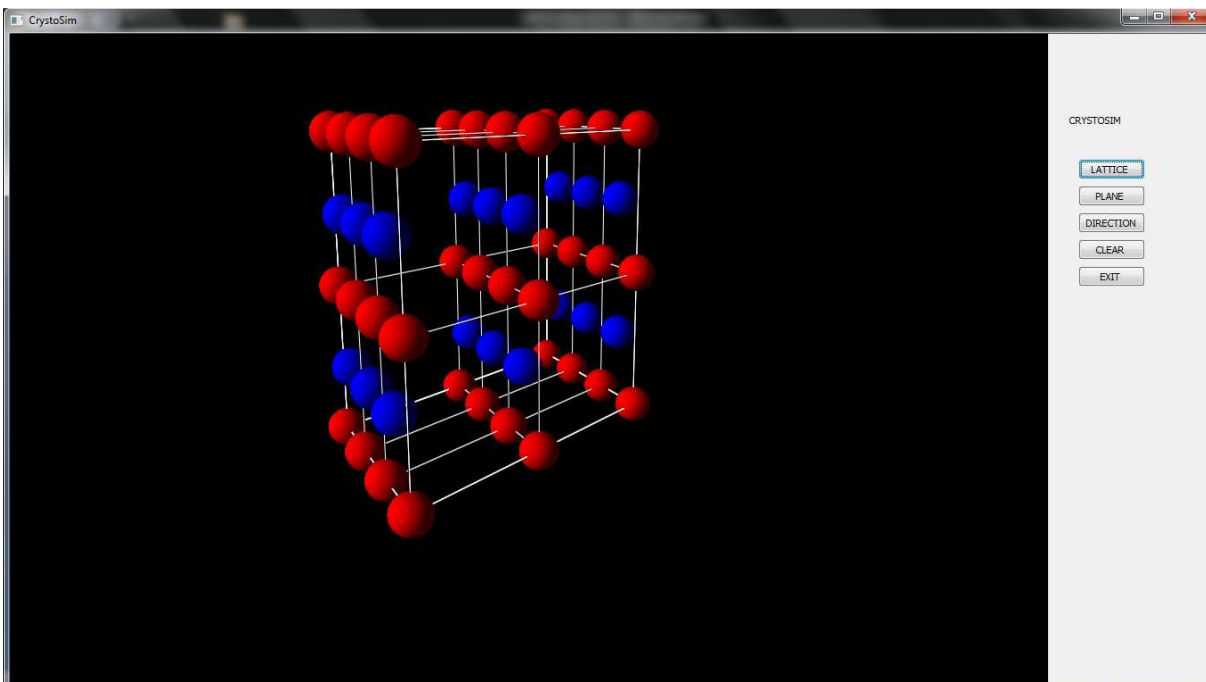
$a=1.1$  ,  $b=2.2$  ,  $c=3.1$  ,  $\beta=75$  ,  $p=50$  ,  $ma=2$  ,  $mb=1$  ,  $mc=2$



### END CENTERED:

The values for the example shown below are as follows:

$a=1.1$  ,  $b=2.1$  ,  $c=2.5$  ,  $\beta=60$  ,  $p=50$  ,  $ma=3$  ,  $mb=2$  ,  $mc=2$



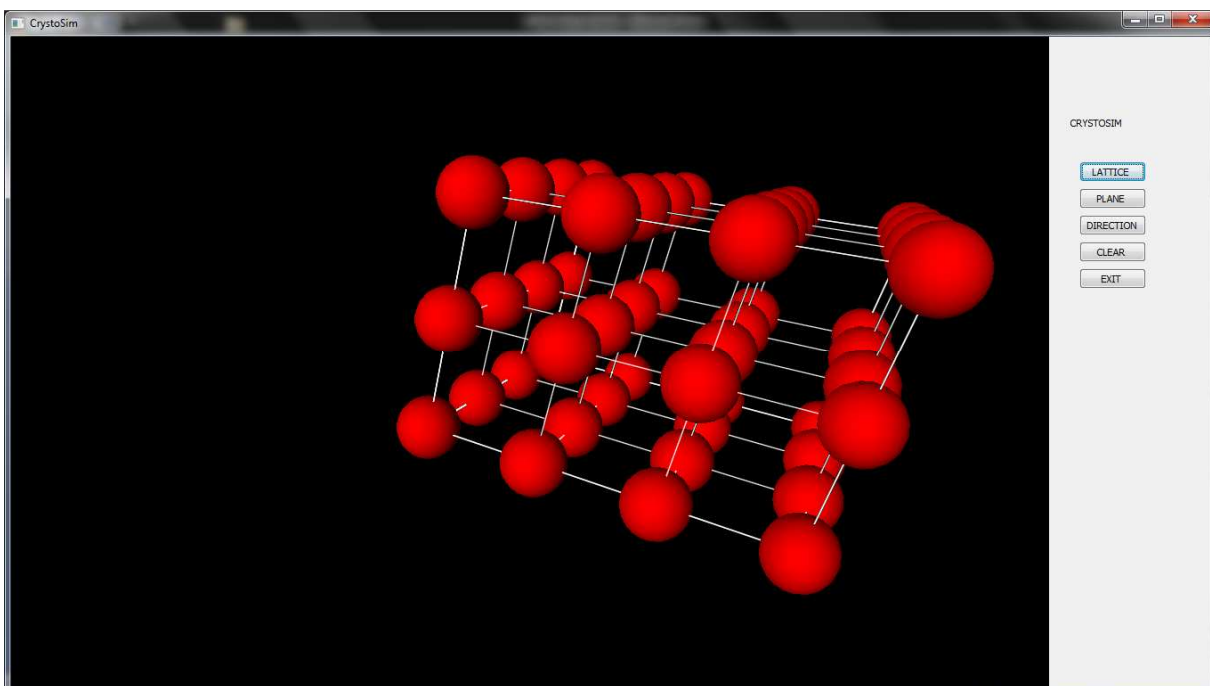
## TRIGONAL:

### PRIMITIVE

Values entered for the example below are as follows:

$a=2.3$  ,  $\alpha=75$  ,  $\beta=50$  ,  $\gamma=90$  ,  $m_a=3$  ,  $m_b=2$  ,  $m_c=3$

In this case  $a=b=c$  and  $\alpha=\beta=\gamma$

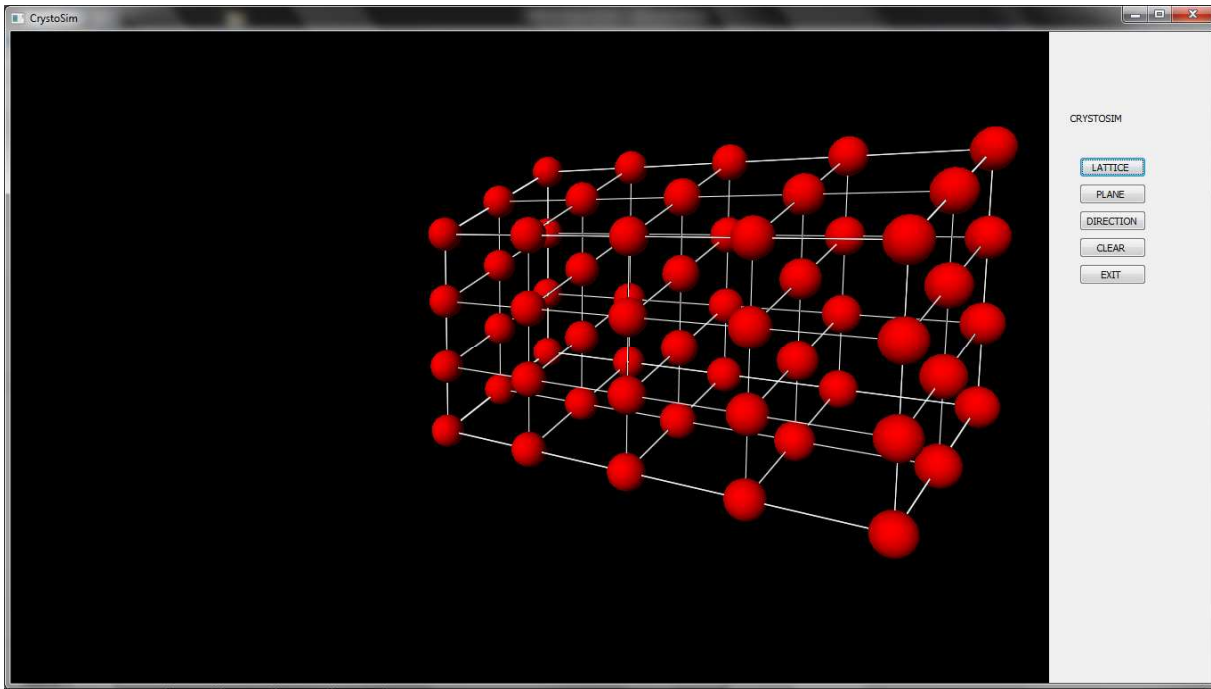


## HEXAGONAL:

### PRIMITIVE

The values for the parameters entered for the example shown below:

$a=1.2$  ,  $c=2.1$  ,  $\beta=90$  ,  $\gamma=120$  ,  $m_a=2$  ,  $m_b=3$  ,  $m_c=4$

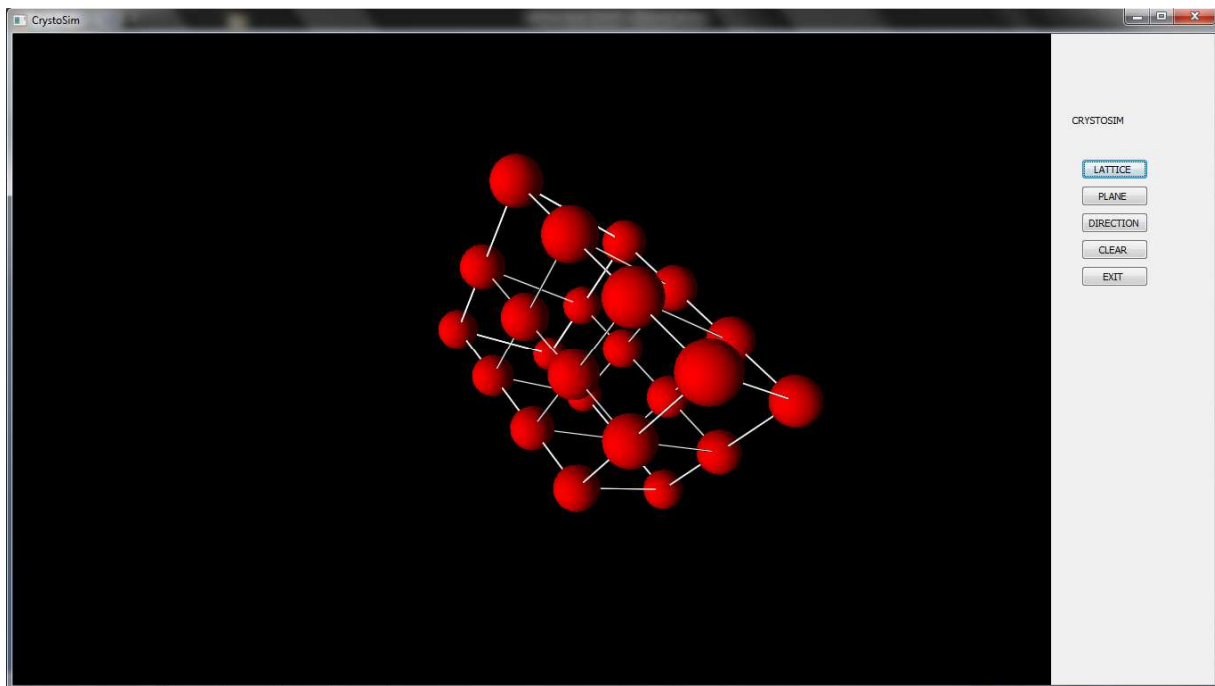


## TRICLINIC:

### PRIMITIVE

Values of the parameters for the example shown below:

$a=1.3$  ,  $b=2.3$  ,  $c=3.1$  ,  $\alpha=45$  ,  $\beta=60$  ,  $\gamma=75$  ,  $p=50$  ,  $ma=3$  ,  $mb=2$  ,  $mc=1$



# ALGORITHM FOR THE GENERATION OF CRYSTALLOGRAPHIC PLANES AND DIRECTIONS

## CRYSTALLOGRAPHIC DIRECTIONS

On being given the miller indices  $h, k$  and  $l$  they are reduced to the simplest form. All the indices are divided by the absolute maximum of the miller indices. Now the values of the miller indices are less than 1.

## FOR CUBIC, TETRAGONAL AND ORTHORHOMBIC CRYSTAL SYSTEMS:

After reducing the miller indices the required projections in the three directions are found by multiplying the indices with the length of the unit cells in each of the three directions. An arrow is plotted from the origin to the obtained points above.

Parameters included in the module include:

$a, b, c$  denotes the length of different sides of the crystal structure in each of the three directions  $x, y$  and  $z$

$h, k, l$  miller indices

## FOR MONOCLINIC, TRIGONAL AND TRICLINIC SYSTEMS

Here to find the length of the projection in the three directions we actually divide the axis of respective length in the ratio of the miller indices:1, i.e. to find the projection on the  $x$  axis we divide the  $x$  axis (from the origin of length  $a$  drawn with appropriate orientation) in the ratio of  $h:1$  and get the projection. Similarly, projections on the  $y$  and  $z$  axis are obtained. Then we draw an arrow from the origin to this point.

Parameters included in the module include:

$a, b, c$  denotes the length of different sides of the crystal structure in each of the three directions  $x, y$  and  $z$

$h, k, l$  miller indices  
 $\alpha, \beta, \gamma$  angles between the faces

### **THE CASE OF HEXAGONAL LATTICE:**

In the hexagonal lattice, the choice of coordinate system is different. We choose three axes  $a_1, a_2$  and  $a_3$  inclined at each other at  $120^\circ$  on the basal plane and a  $z$  axis perpendicular to this basal plane. Parameters taken in this function include

$s$  length of one side of the hexagon

$u, v, t$  and  $w$  the miller indices

Here we just input the miller indices across the two basal plane axes  $a_1$  and  $a_2$  represented by  $u$  and  $v$  and the miller index across  $z$  represented by  $w$ . After reducing them to the simplest form to find the length of the projection in the three directions we actually divide the axis of respective length in the ratio of the miller indices:1, i.e. to find the projection on the  $x$  axis we divide the  $x$  axis (from the origin of length  $a$  drawn with appropriate orientation) in the ratio of  $u:1$  and get the projection. Then the arrow is drawn from the origin to the required point obtained from the projection. In case any of the miller index is negative the required point is obtained by rotating the obtained point by  $\pi$  radians.

### **CRYSTALLOGRAPHIC PLANE:**

On being given the miller indices  $h, k$  and  $l$  they are reduced to the simplest form. This is done by finding out the hcf between  $h$  and  $k$  and also between  $k$  and  $l$ , then the hcf of the two obtained nos is found. All the three indices are then divided by the combined hcf. Then  $h, k$  and  $l$  are made equal to their reciprocals.

### **FOR CUBIC, TETRAGONAL AND ORTHORHOMBIC CRYSTAL SYSTEMS:**

After reducing the miller indices the required projections are found by multiplying the indices with the length of the unit cells in each of the three directions. Then the points of the planes are located for the different cases and a plane is drawn using the convex function is drawn.

Parameters included in the module include:

$a, b, c$  denotes the length of different sides of the crystal structure in each of the three directions  $x, y$  and  $z$

$h, k, l$  miller indices

## **FOR MONOCLINIC, TRIGONAL AND TRICLINIC SYSTEMS**

Here to find the length of the projection in the three directions we actually divide the axis of respective length in the ratio of the miller indices:1 ,i.e. to find the projection on the x axis we divide the x axis (from the origin of length a drawn with appropriate orientation)in the ratio of h:1 and get the projection. Similarly, points of the plane on the y and z axis are obtained and the required plane is constructed.

Parameters included in the module include:

a,b,c                                    denotes the length of different sides of the crystal structure in each of the three directions x , y and z

h1,k,l1                                    miller indices

alpha,beta,gamma                    angles between the faces

## **THE CASE OF HEXAGONAL LATTICE:**

In the hexagonal lattice , the choice of coordinate system is different. We choose three axis a1,a2 and a3 inclined at each other at 120 on the basal plane and a z axis perpendicular to this basal plane. Parameters taken in this function include

s    :    length of one side of the hexagon

u,v,t and w                            :    the miller indices

Here we just input the miller indices across the two basal plane axis a1 and a2 represented by u and v and the miller index across z represented by w. After reducing them to the simplest form and equating them to their reciprocals to find the length of the projection in the three directions we actually divide the axis of respective length in the ratio of the miller indices:1, i.e. to find the projection on the x axis we divide the x axis (from the origin of length a drawn with appropriate orientation)in the ratio of u:1 and get the point .. In case any of the miller index is negative the required point is obtained by rotating the obtained point by pi radians. Then the required plane is constructed.

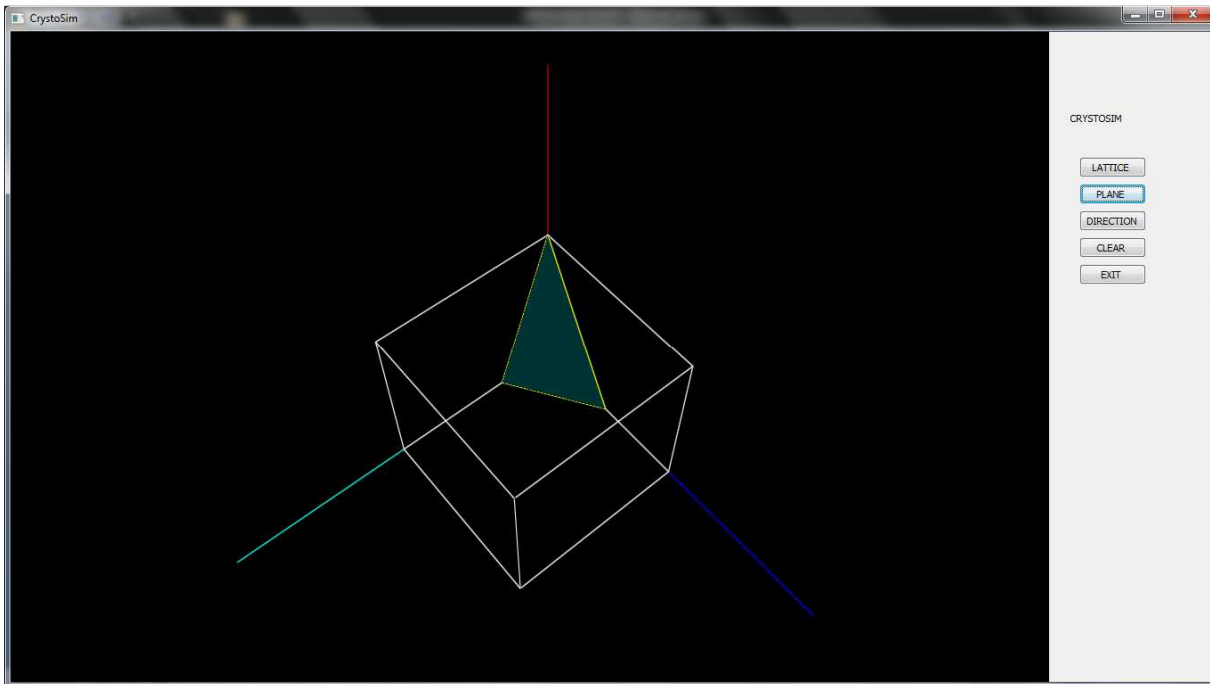
# DEMONSTRATION

## CRYSTALLOGRAPHIC PLANES:

### CUBIC

For the example shown below the parameters entered are :

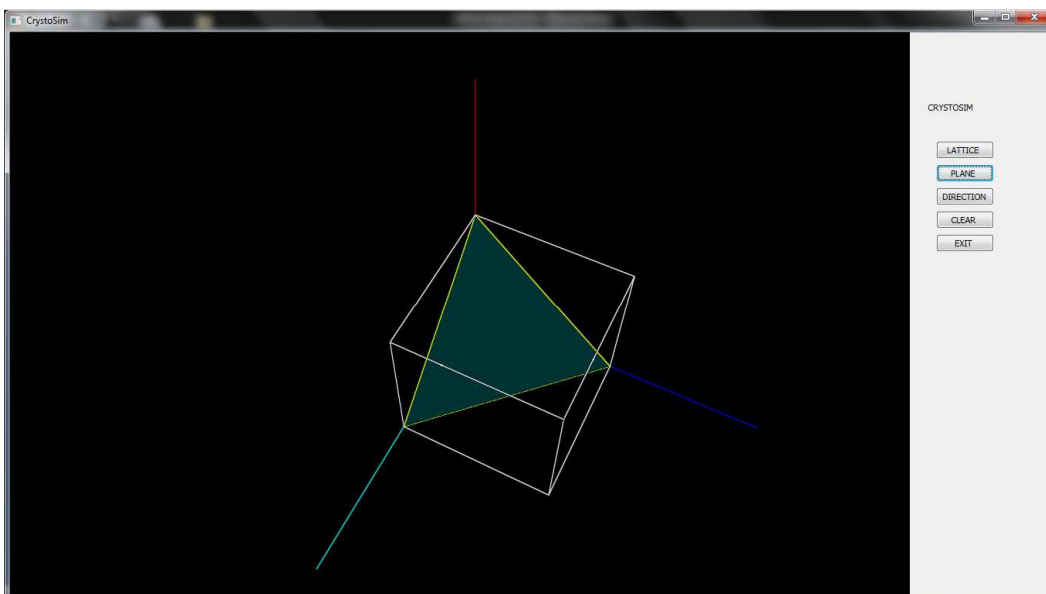
$a=1$  ,  $h=2$  ,  $k=1$ ,  $l=3$



### TETRAGONAL:

For the example shown below the parameters entered are:

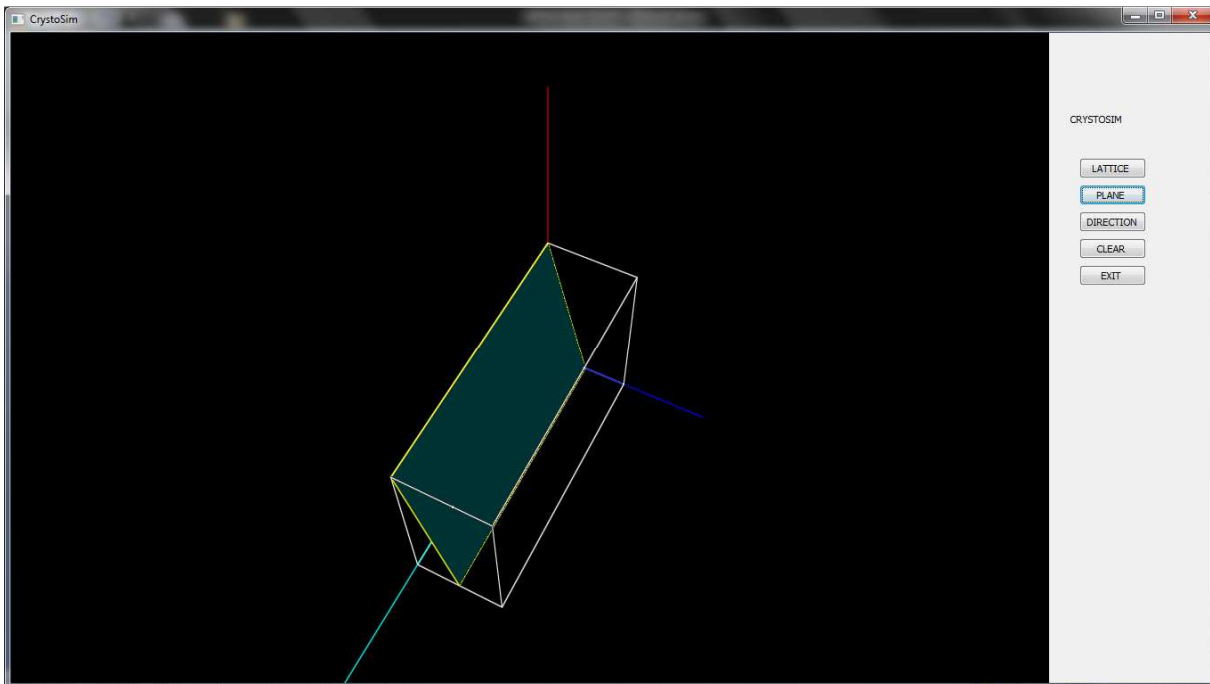
$a=1$  ,  $c=1$ ,  $h=2$  ,  $k=2$  ,  $l=2$



## ORTHORHOMBIC:

The parameters entered for the example below are:

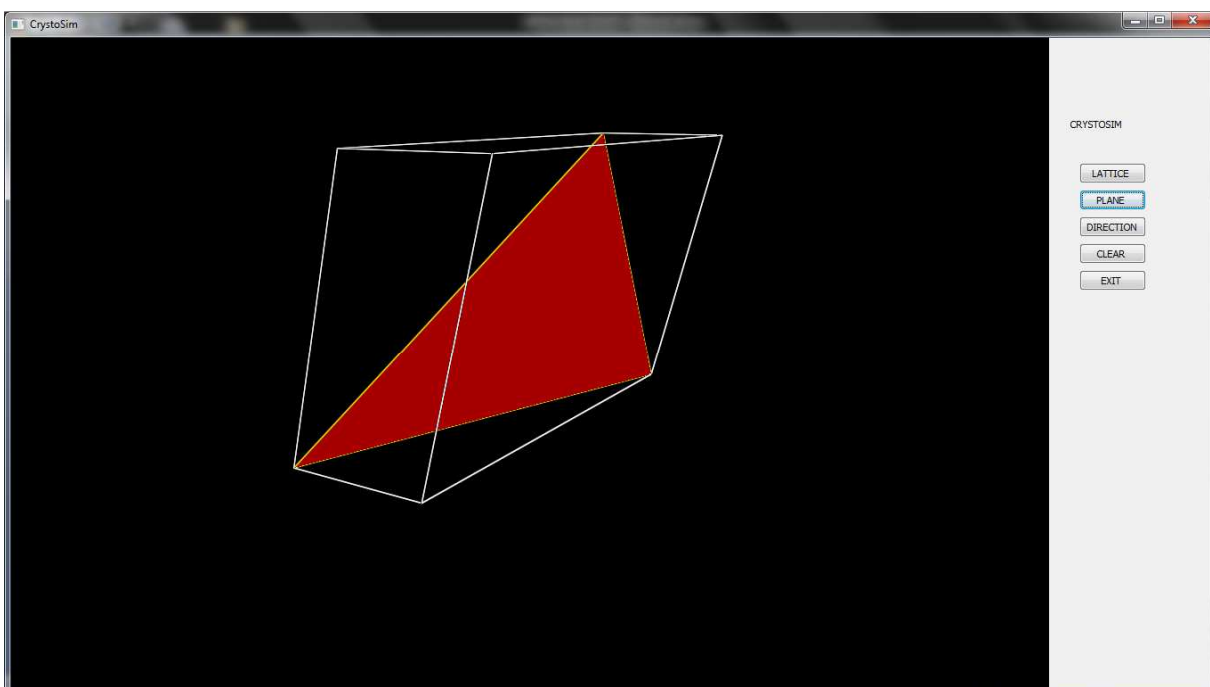
$a=1, b=2, c=3, h=2, k=1, l=0$



## MONOCLINIC:

The parameters entered for the example shown below:

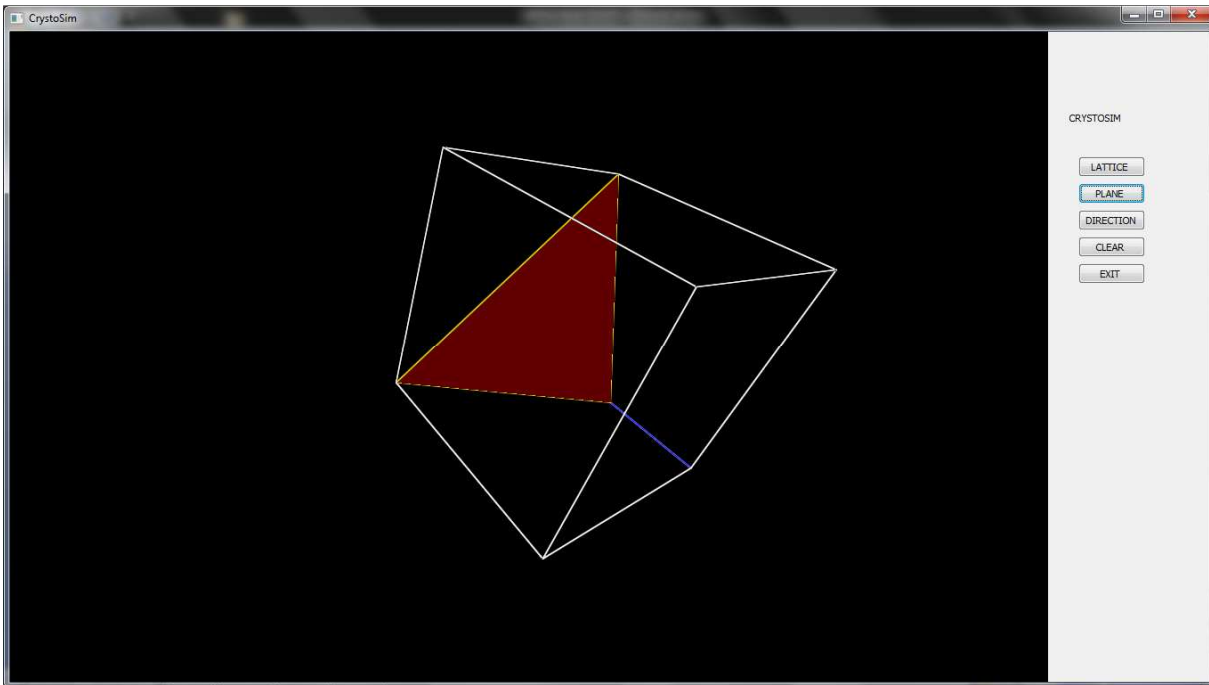
$a=1, b=2, c=3, \alpha=90, \beta=75, h=1, k=1, l=1$



## TRIGONAL:

The parameters entered for the example case below are:

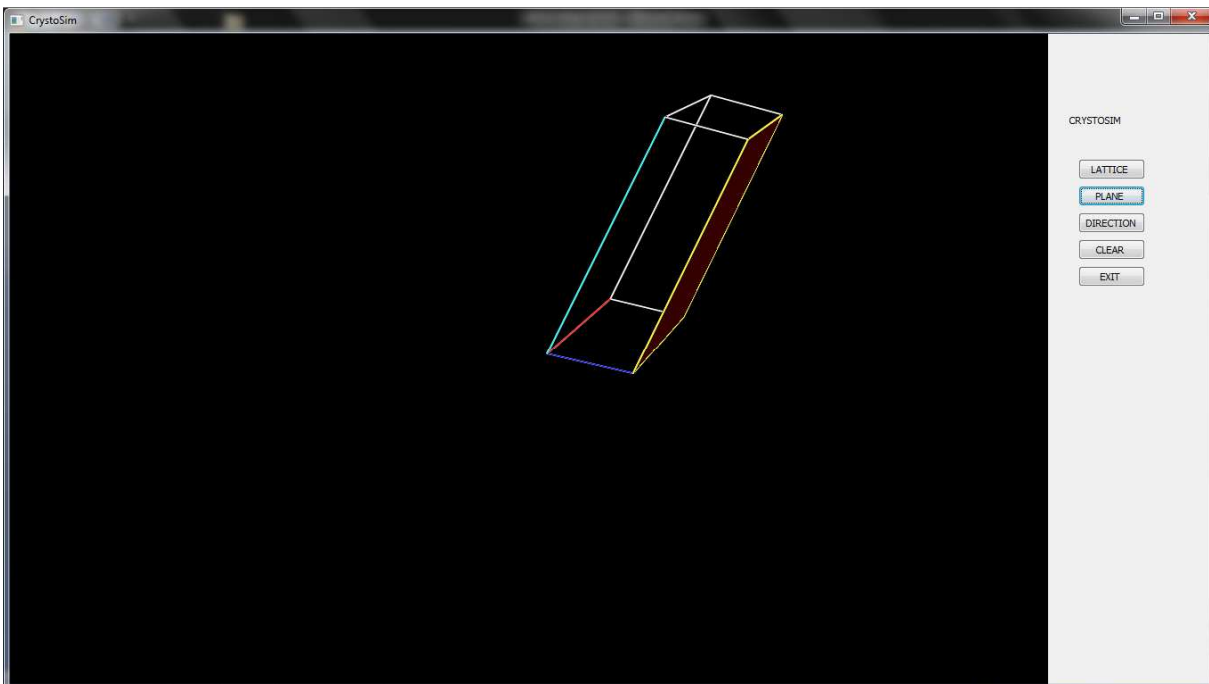
$a=1$ ,  $\alpha=60$ ,  $l=1$ ,  $w=2$ ,  $h=1$



## TRICLINIC :

Parameters entered for the example case below are as follows:

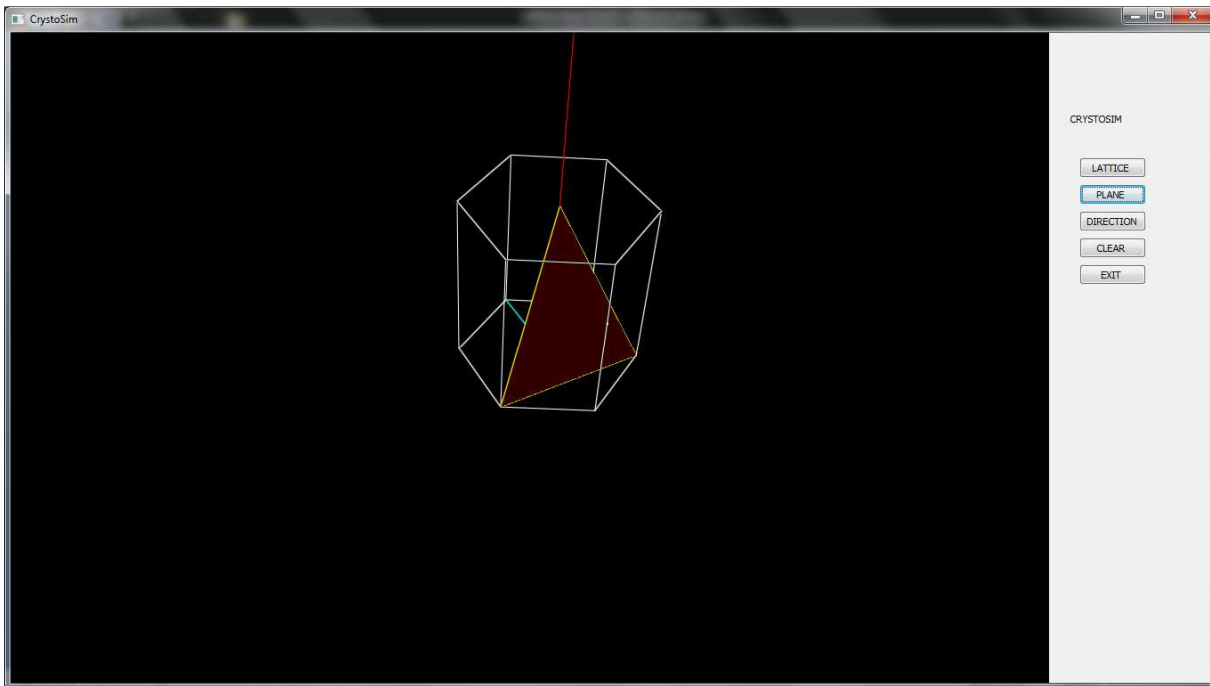
$a=1$ ,  $b=2$ ,  $c=3$ ,  $\alpha=56$ ,  $\beta=65$ ,  $\gamma=76$ ,  $h=1$ ,  $k=1$ ,  $l=0$



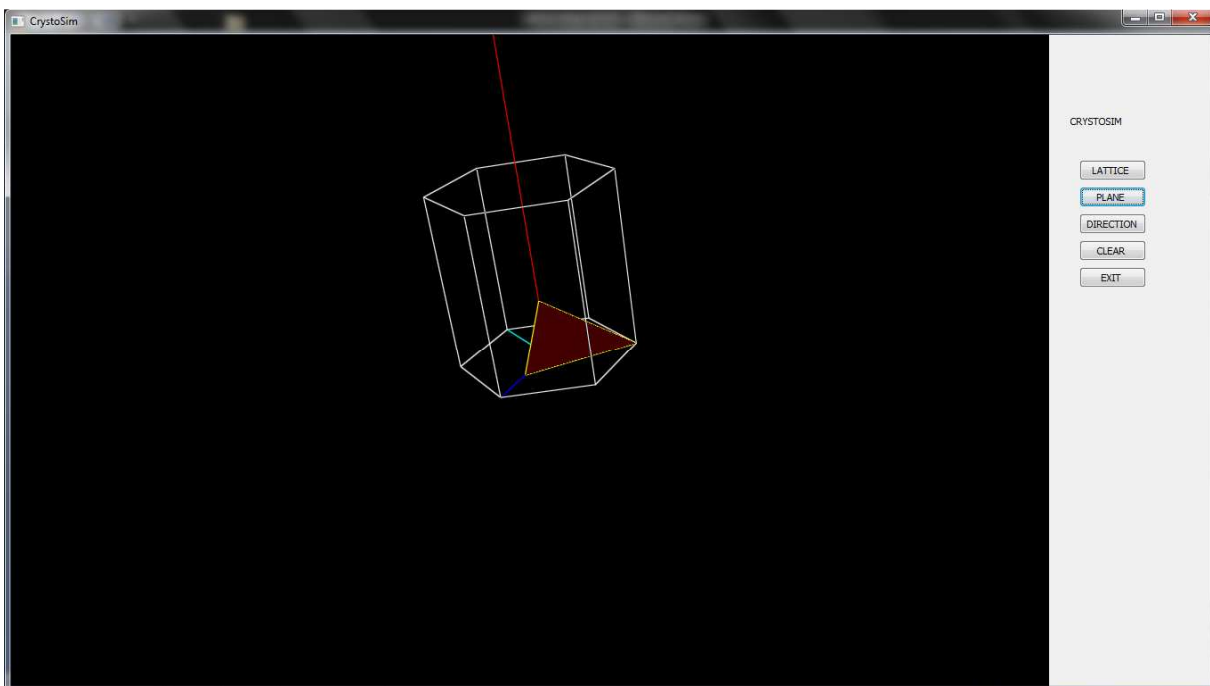
## HEXAGONAL:

Parameters entered for the example case below are as follows:

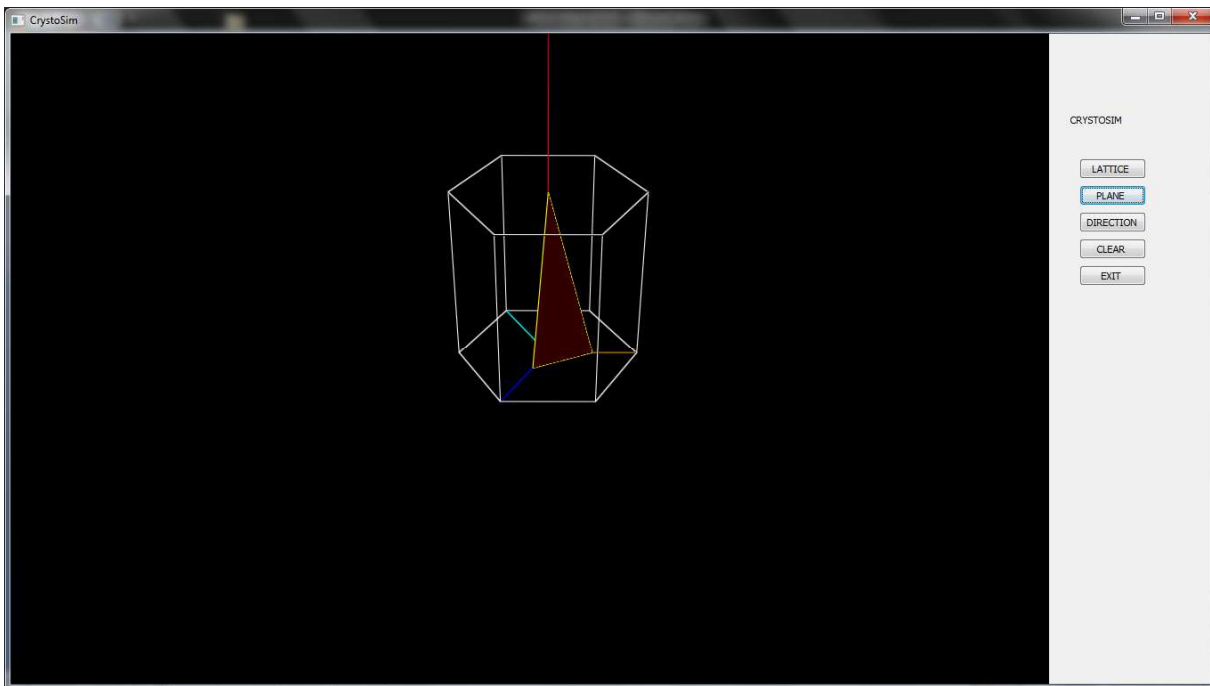
$s=1, u=1, v=1, w=1$



Miller indices 2 1 3



Miller indices 3 2 1

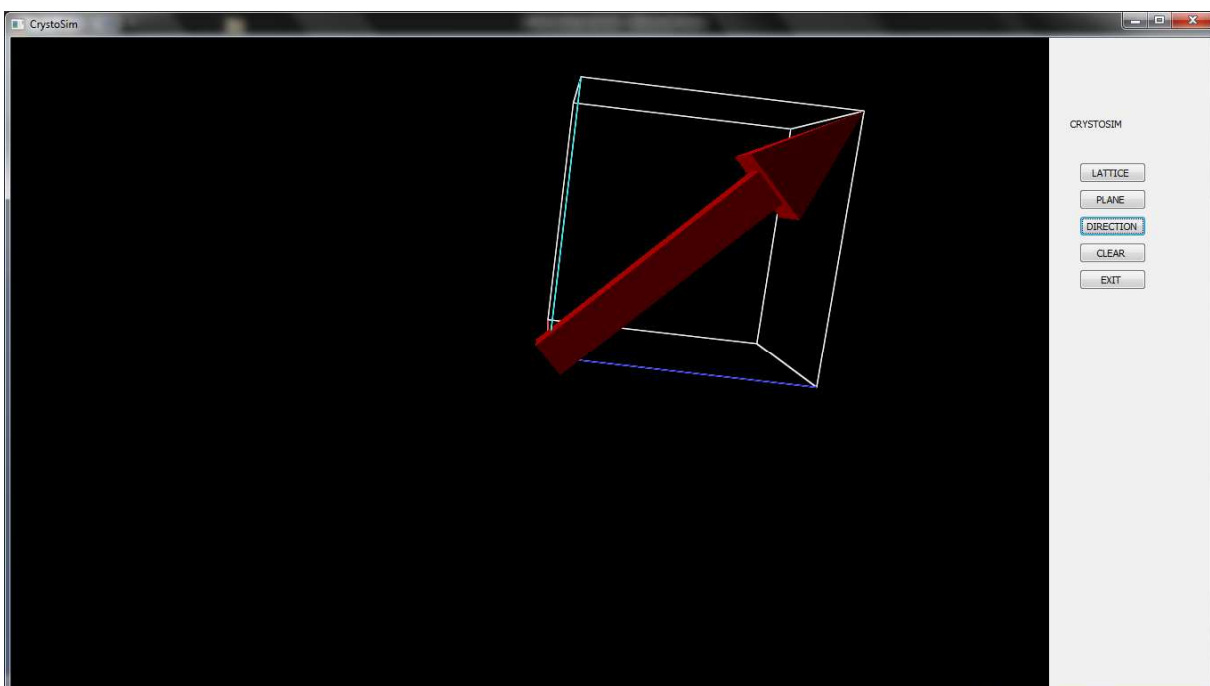


## CRYSTALLOGRAPHIC PLANES :

### CUBIC:

For the example shown below the parameters entered are :

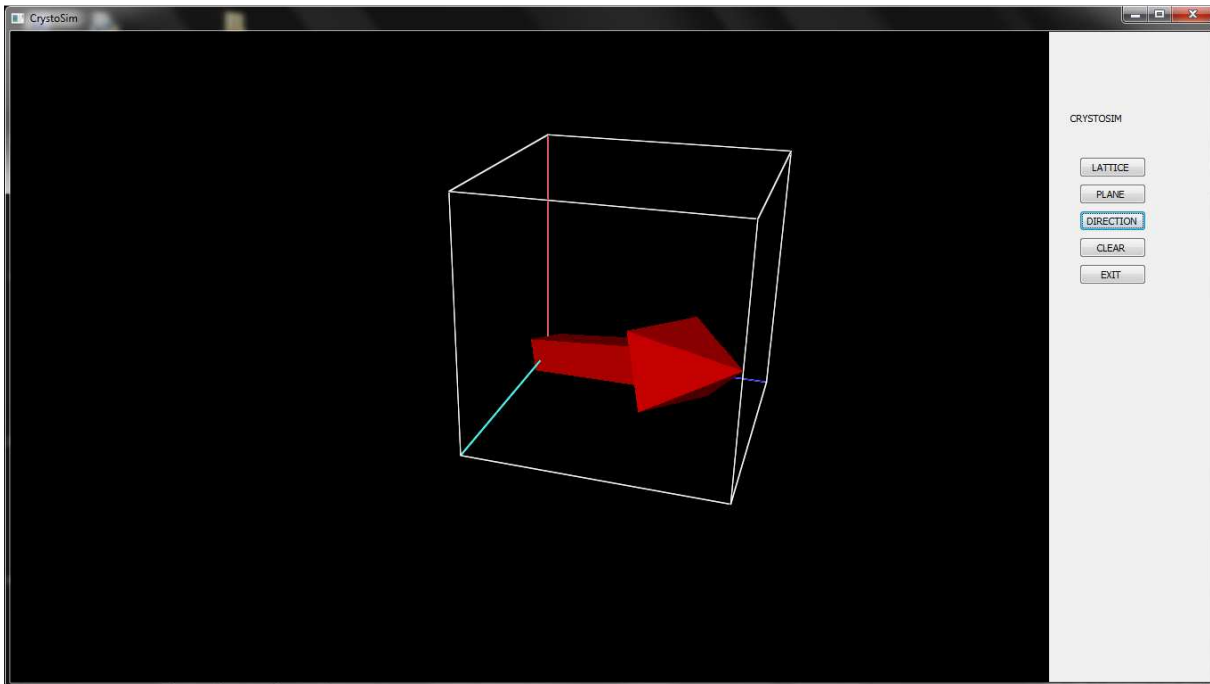
$a=1$  ,  $h=1$  ,  $k=0$  ,  $l=1$



## TETRAGONAL:

For the example shown below the parameters entered below are :

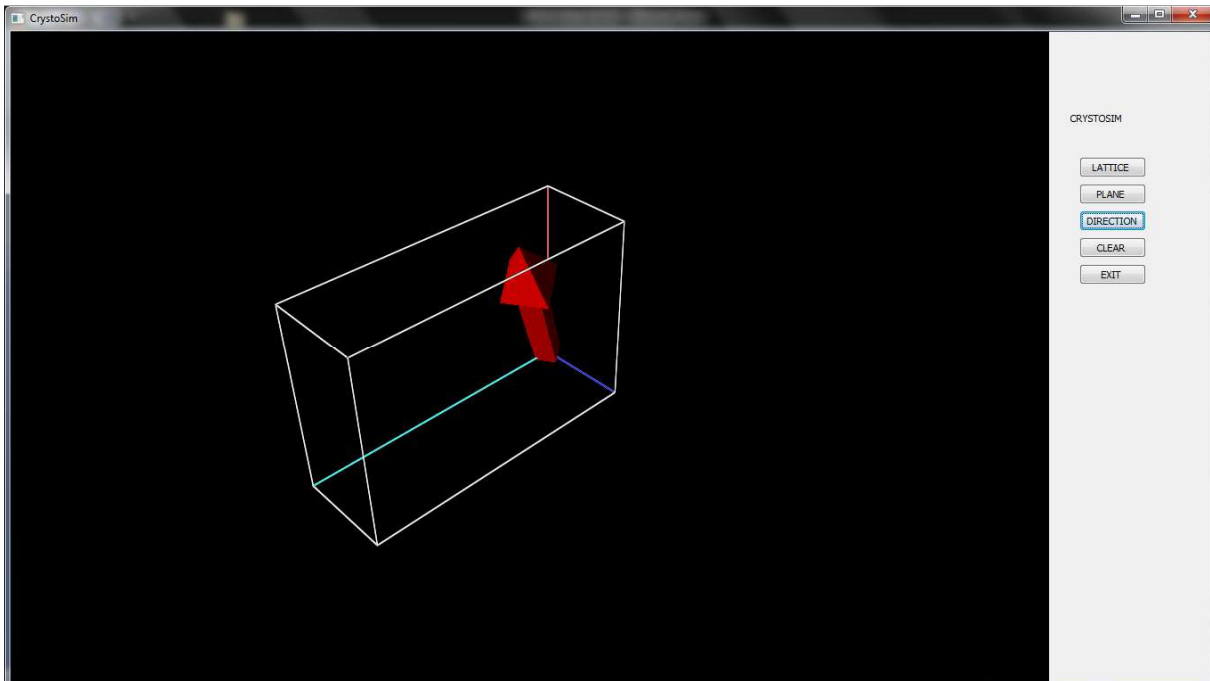
$a=1$  ,  $b=1$  ,  $h=2$ ,  $k=1$ ,  $l=2$



## ORTHORHOMBIC

For the example shown below the parameters entered below are :

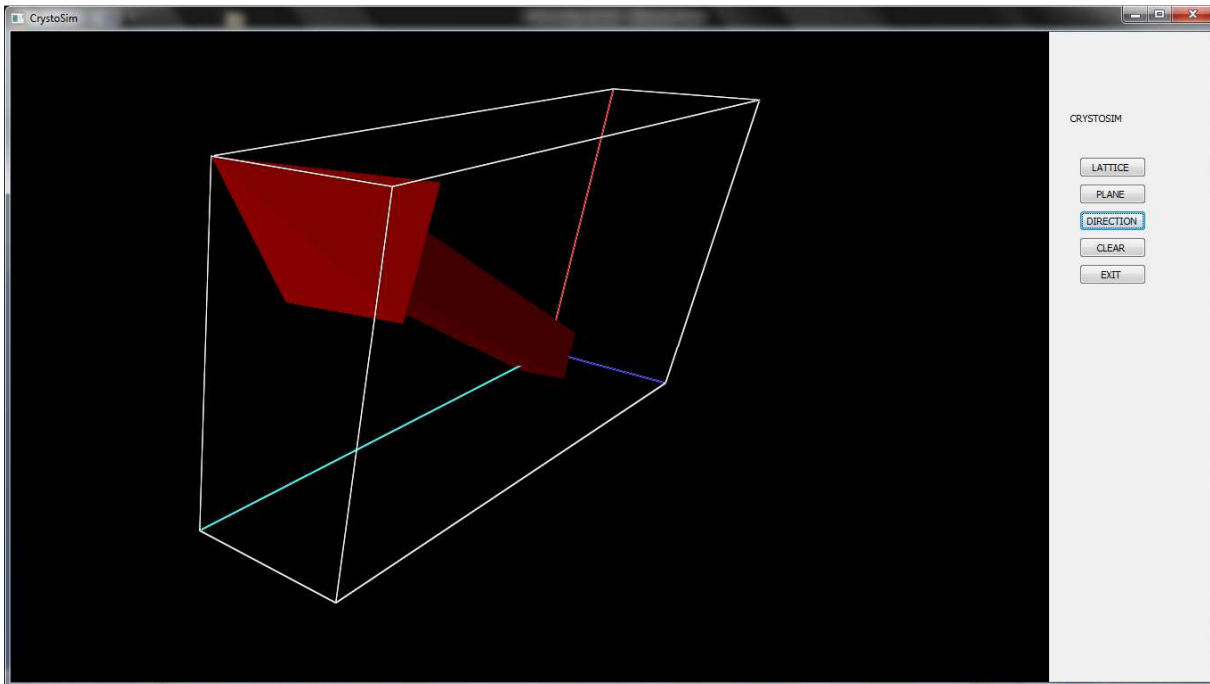
$a=1$  ,  $b=2$ ,  $c=3$  ,  $h=2$ ,  $k=3$ ,  $l=1$



## MONOCLINIC:

Parameters as entered for the example shown below:

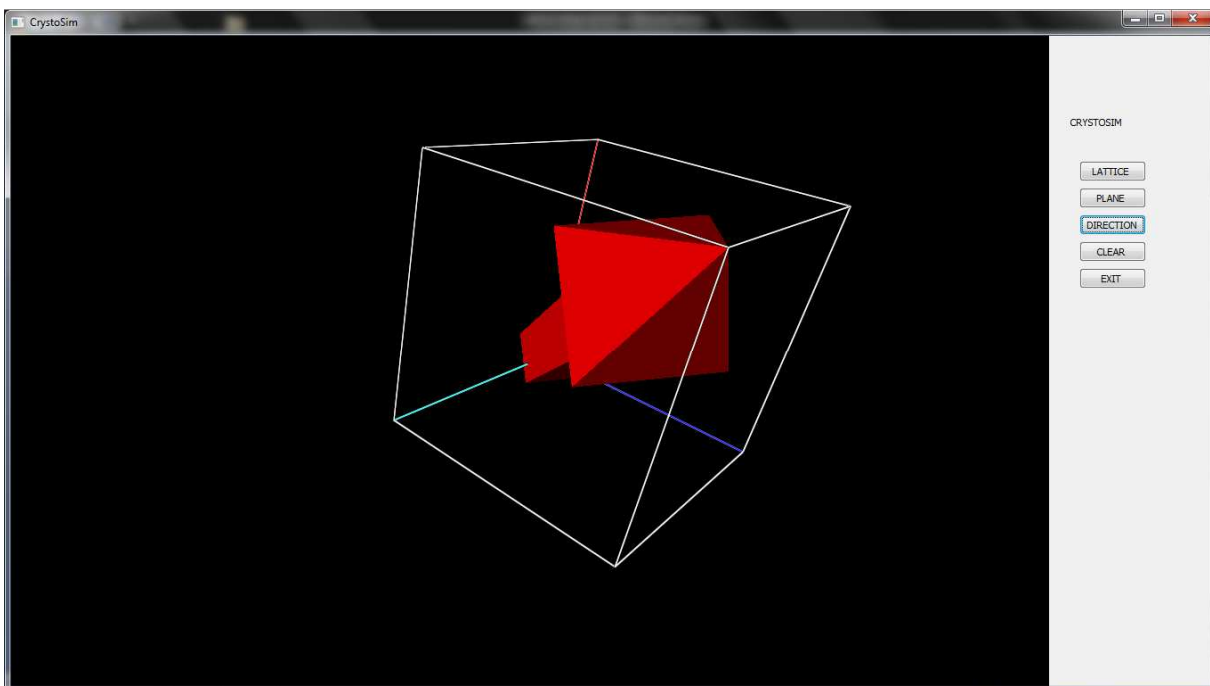
$a=1, b=2, c=3, \alpha=90, \beta=75, h=0, k=1, l=1$



## TRIGONAL

Values entered for the example below are :

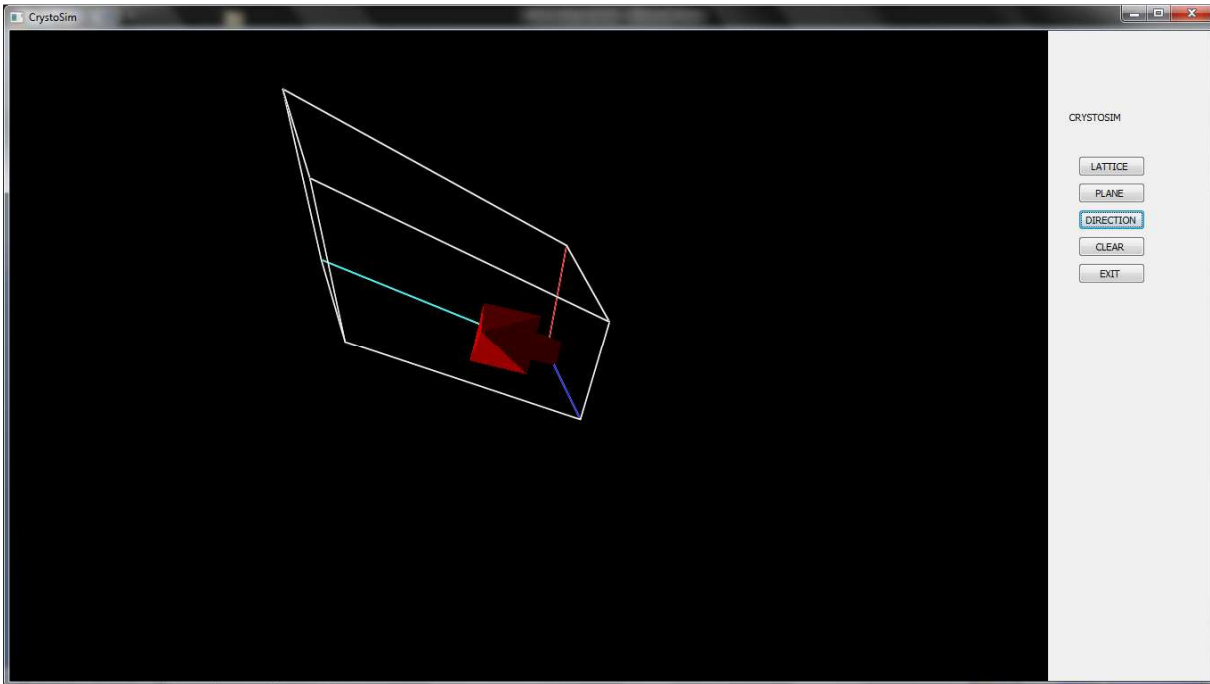
$a=1, \alpha=75, h=1, k=1, l=1$



## TRICLINIC:

For the example below , the value of the parameters entered are :

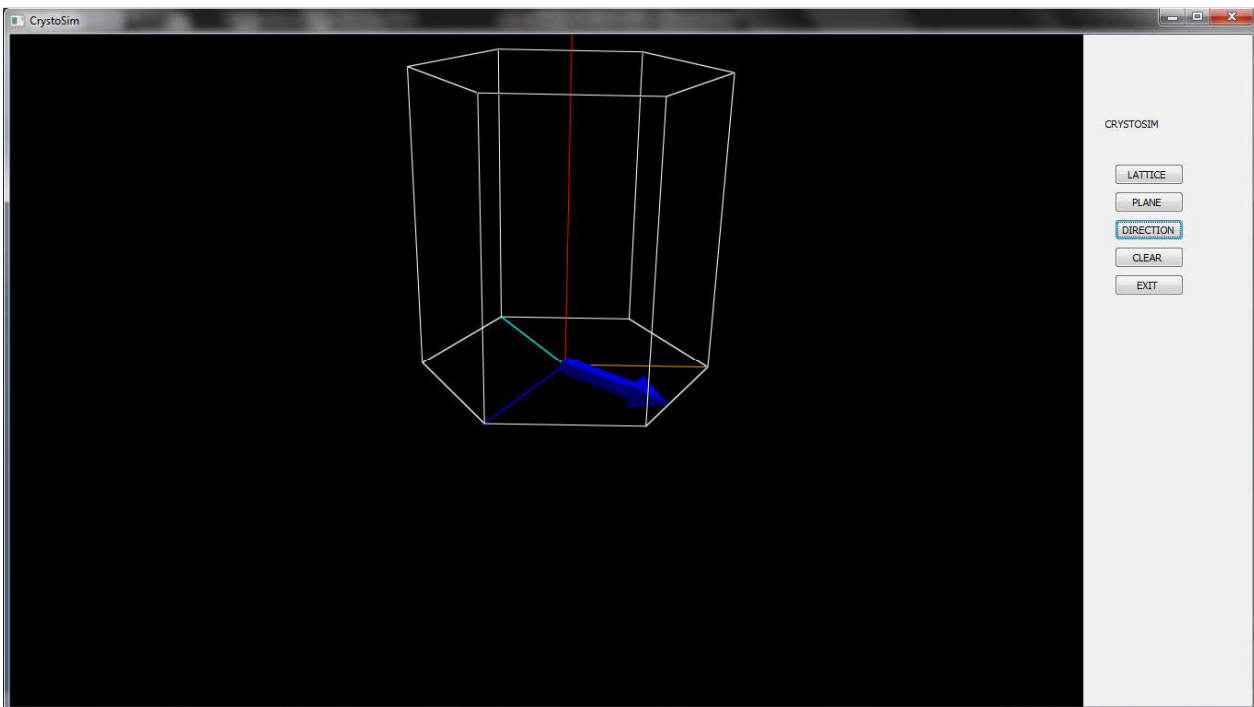
$a=1$ ,  $b=2$ ,  $c=3$ ,  $\alpha=45$ ,  $\beta=75$ ,  $\gamma=86$ ,  $h=2$ ,  $k=1$ ,  $l=1$



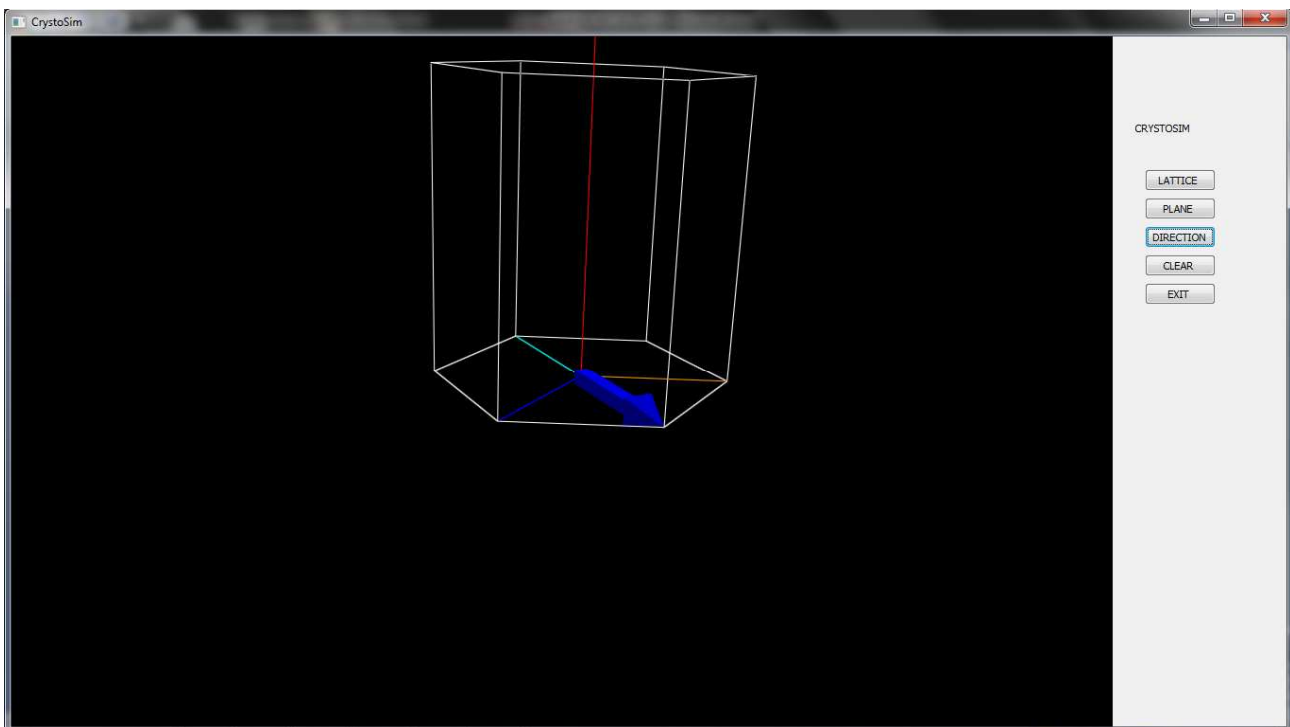
## HEXAGONAL:

Values entered for the case shown below are :

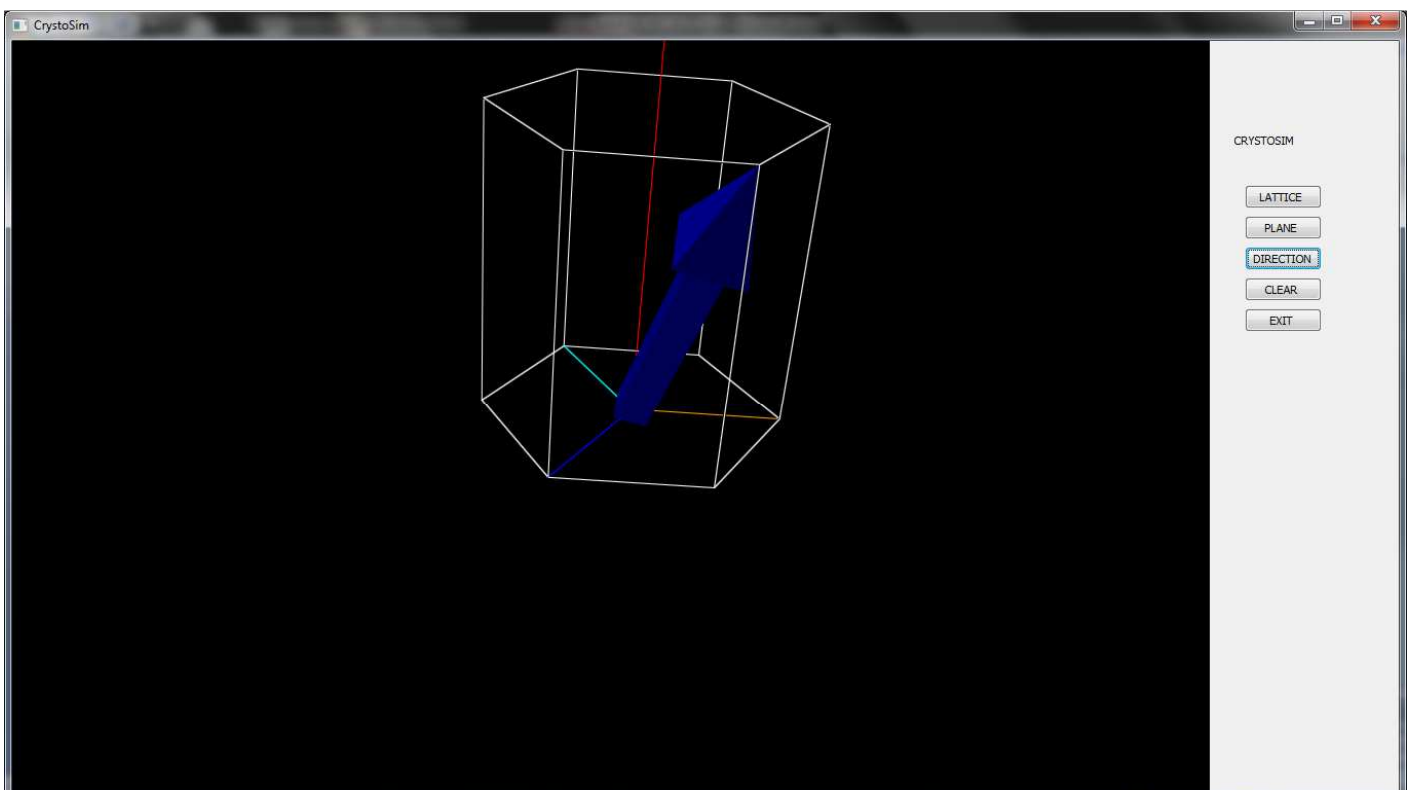
$s=0, u=2, v=3, t=0$



For the miller indices 1 1 0



For the miller indices 1 1 1



## **PROBLEMS FACED:**

Earlier I used the basic coordinate system approach to locate the points in the lattice but found problems with the crystal systems where angle between the faces were not 90. So used the vector approach and found it very useful and handy even while drawing directions and planes.

## **FURTHER IMPROVEMENTS:**

- ▶ This project can be extended to find the intersection of two planes in the lattice system.
- ▶ The project could also be extended to simulate voids, dislocations, symmetry and even effect of stress fields on dislocations and atoms interactively.

REFERENCES:

<http://www.vpython.org/>

<http://wiki.wxpython.org/AnotherTutorial>

<http://www.wxpython.org/tut-part1.php>

Learning Python by Mark Lutz