Computer Graphics

Lecture 11

Complex Numbers

The introduction of complex numbers in the 16th century made it possible to solve the equation $x^2 + 1 = 0$.

A complex number is a number that can be expressed in the form

$a + bi$,

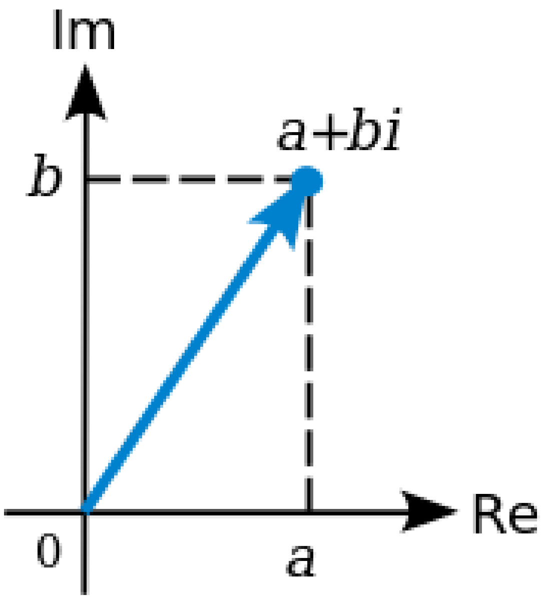
where a and b are real numbers and i is the imaginary unit which satisfies the equation $i^2 = -1$

In other words *a* is the *real part* and *b* is the *imaginary part* of the complex number.

Visualizing Complex Number

Complex numbers extend the concept of the one-dimensional number line to the two-dimensional complex plane

Let horizontal axis use for the real part and the vertical axis use for the imaginary part



A complex number can be visually represented as a pair of numbers (a, b) forming a vector on a diagram called an Argand diagram, representing the complex plane. "Re" is the real axis, "Im" is the imaginary axis, and i is the imaginary unit which satisfies i2 = −1.

https://en.wikipedia.org/wiki/Complex_number

Complex Plane

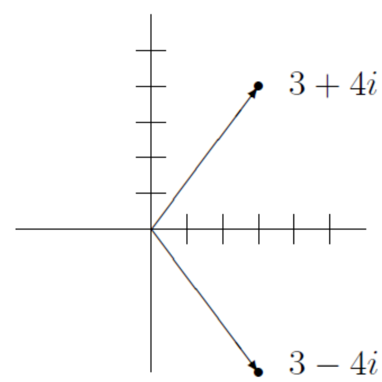
A complex number z is given by a pair of real numbers x and y and is written in the form $z = x + iy$, *where i satisfies* $i^2 = -1$.

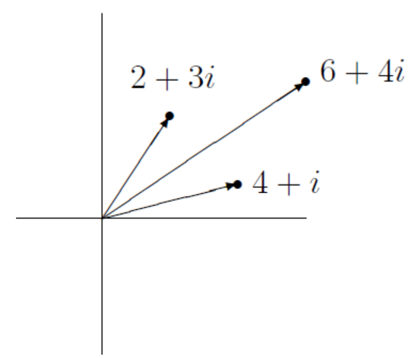
The real number 1 is represented by the point (1, 0)

The complex number *i* is represented by the point (0, 1)

In the Argand diagram (complex number represent in a plane)

- x-axis is called the "real axis"
- y-axis is called the "imaginary axis"





Complex Operations

If $z_1 = x_1 + iy_1$ and $z_2 = x_2 + iy_2$,

Addition:

$$z_1 + z_2 = (x_1 + x_2) + i(y_1 + y_2)$$

Multiplication:

$$z_1 z_2 = (x_1 x_2 - y_1 y_2) + i(x_1 y_2 + x_2 y_1)$$

Polar Representation of Complex Numbers

The relation between the rectangular coordinates (x, y) and the polar coordinates (r,θ)

$$x = r\cos(\theta) \qquad \text{and} \qquad y = r\sin(\theta)$$

$$r = \sqrt{x^2 + y^2} \qquad \text{and} \qquad \theta = \arctan(y/x)$$

Thus, for the complex number z = x + iy, we can write

$$z = r(\cos\theta + i\sin\theta)$$

Polar Representation of Complex Numbers

Using Euler Formula

e^x can be expressed as the following *power series*

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots + \frac{x^n}{n!} + \ldots$$

For any complex number z, we *define* e^z by the power series:

$$e^z = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \ldots + \frac{z^n}{n!} + \ldots$$

In particular,

$$\begin{aligned} e^{i\theta} &= 1 + i\theta + \frac{(i\theta)^2}{2!} + \frac{(i\theta)^3}{3!} + \ldots + \frac{(i\theta)^n}{n!} + \ldots \\ &= 1 + i\theta - \frac{\theta^2}{2!} - \frac{i\theta^3}{3!} + \frac{\theta^4}{4!} + \ldots \end{aligned}$$

Polar Representation of Complex Numbers

The functions $\cos(\theta)$ and $\sin(\theta)$ can also be written as power series:

$$\cos(\theta) = 1 - \frac{\theta^2}{2} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \ldots + \frac{(-1)^n\theta^{2n}}{(2n)!} + \ldots$$

$$\sin(\theta) = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \ldots + \frac{(-1)^n\theta^{2n+1}}{(2n+1)!} \pm \ldots$$

Thus

(the power series for $e^{i\theta}$) = (the power series for $\cos(\theta)$)+$i\cdot$(the power series for $\sin(\theta)$)

This is the Euler Formula:

$$e^{i\theta} = \cos(\theta) + i\sin(\theta)$$

$$e^{i\pi/2} = i, \qquad e^{\pi i} = -1 \qquad \text{and} \qquad e^{2\pi i} = +1$$

Given $z = x + iy$, then z can be written in the form $z = re^{i\theta}$, where

$$r = \sqrt{x^2 + y^2} = |z| \qquad \text{and} \qquad \theta = \tan^{-1}(y/x)$$

Quaternions

Consider 3 dimensional vector space v with the basis a, b, c

v = a + b + c (the simplest linear combination of every basis vector)

then

$$
\begin{aligned}
v^2 &= v(v) = (a+b+c)*(a+b+c) \\
&= a(a+b+c)+b(a+b+c)+c(a+b+c) \\
&= a^2+ab+ac+ba+b^2+bc+ca+cb+c^2 \\
&= (a^2+b^2+c^2)+ab+ac+ba+bc+ca+cb
\end{aligned}
$$

Quaternions

As the Complex number of written as a + bi

the quaternions of R^4 written as a+bi+cj+dk

where we have suppressed 1 = (1, 0, 0, 0), and

$i^2 = j^2 = k^2 = ijk = -1$;

multiplication is associative, so that

$ij = k$; $jk = i$; $ki = j$;

$ji = -k$; $kj = -i$; $ik = -j$

Quaternions

Notations:

<1,2> $\Rightarrow$ $1+2i$

<1,2,3,4> $\Rightarrow$ $1+2i+3j+4k$

Additions:

$(1+2i+3j+4k) + (j+k) = 1+2i+4j+5k$

In general

$(a+bi+cj+dk)+(e+fi+gj+hk) = (a+e)+(b+f)i+(c+g)j+(d+h)k$

Quaternions Multiplication

×	1	*i*	*j*	*k*
1	1	*i*	*j*	*k*
i	*i*	−1	*k*	−*j*
j	*j*	−*k*	−1	*i*
k	*k*	*j*	−*i*	−1

Consider two elements given by

$a_1 + b_1 i + c_1 j + d_1 k$ and $a_2 + b_2 i + c_2 j + d_2 k$,

Hamilton product of two elements is

$$
\begin{aligned}
= \; & a_1 a_2 + a_1 b_2 i + a_1 c_2 j + a_1 d_2 k \\
& + b_1 a_2 i + b_1 b_2 i^2 + b_1 c_2 ij + b_1 d_2 ik \\
& + c_1 a_2 j + c_1 b_2 ji + c_1 c_2 j^2 + c_1 d_2 jk \\
& + d_1 a_2 k + d_1 b_2 ki + d_1 c_2 kj + d_1 d_2 k^2
\end{aligned}
$$

$$
\begin{aligned}
= \; & a_1 a_2 - b_1 b_2 - c_1 c_2 - d_1 d_2 \\
& + (a_1 b_2 + b_1 a_2 + c_1 d_2 - d_1 c_2) i \\
& + (a_1 c_2 - b_1 d_2 + c_1 a_2 + d_1 b_2) j \\
& + (a_1 d_2 + b_1 c_2 - c_1 b_2 + d_1 a_2) k.
\end{aligned}
$$

More on Complex numbers

$i^1 = i$

$i^2 = -1$

$i^3 = -i$

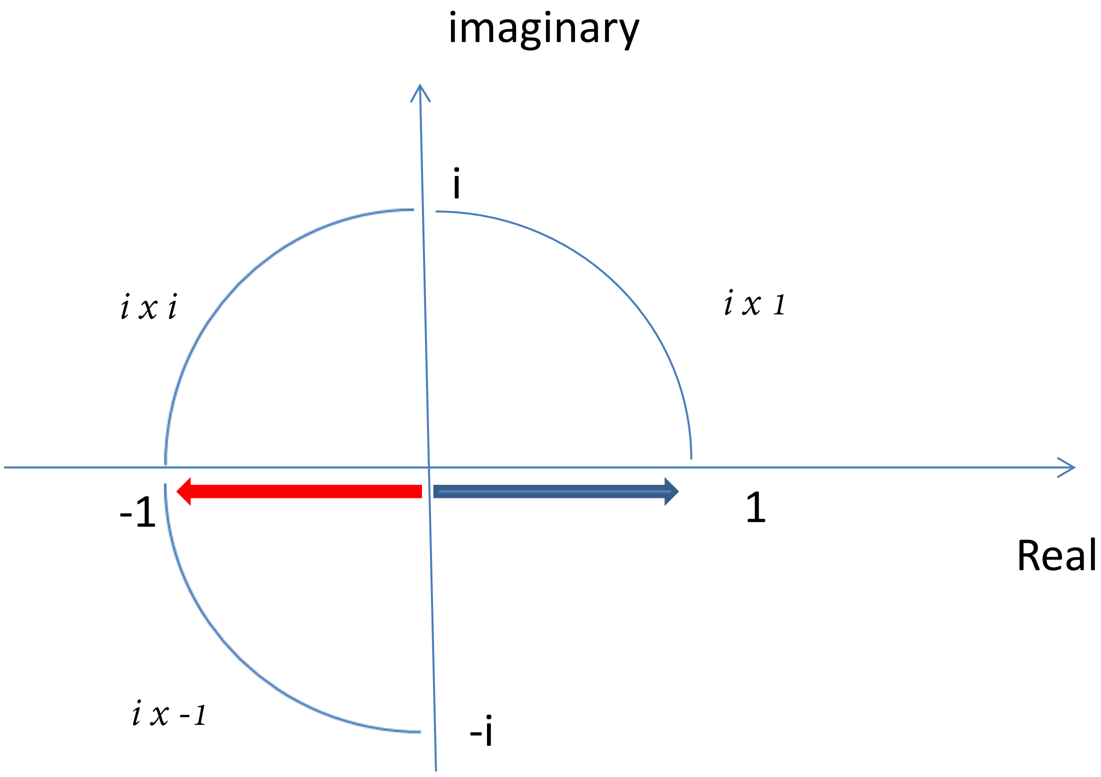
$i^4 = 1$

$i^5 = i$

$i^6 = -1$

$i^7 = -i$

$i^8 = 1$



Numbers are 2-dimentional meaning numbers have hidden dimension

Exercise

Recall

Thus

(the power series for $e^{i\theta}$) = (the power series for $\cos(\theta)$)+$i\cdot$(the power series for $\sin(\theta)$)

This is the Euler Formula:

$$e^{i\theta} = \cos(\theta) + i\sin(\theta)$$

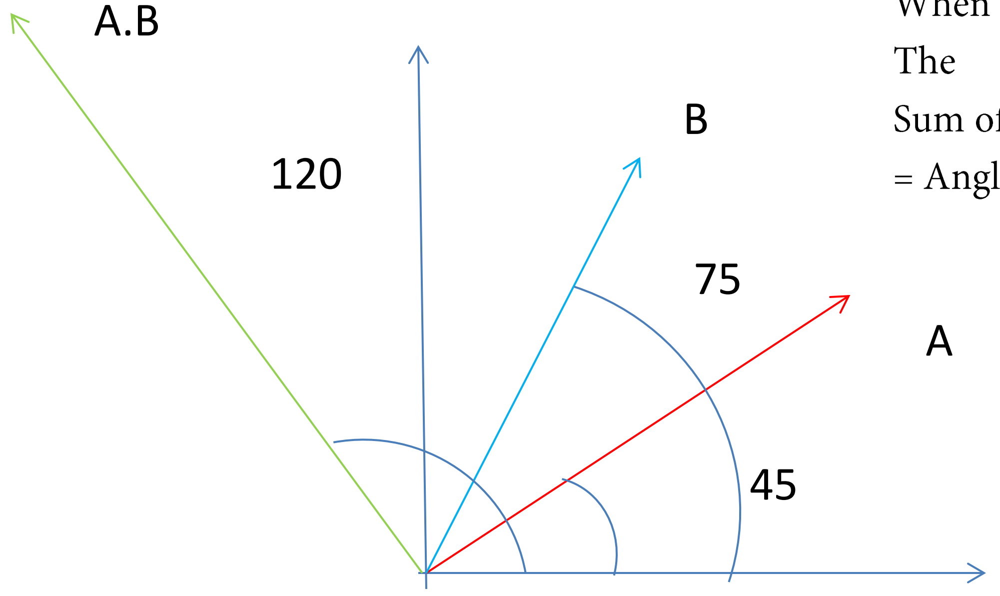
$$e^{i\pi/2} = i, \qquad e^{\pi i} = -1 \qquad \text{and} \qquad e^{2\pi i} = +1$$

Given $z = x + iy$, then z can be written in the form $z = re^{i\theta}$, where

$$r = \sqrt{x^2 + y^2} = |z| \qquad \text{and} \qquad \theta = \tan^{-1}(y/x)$$

Problem	Angle 1	Angle 2	Results	Result Angle
(4+3i).i	36.9	-90	-3+4i	-53.1
(4+3i).2i	36.9	-90	-6.8i	-53.1
(4+3i).(4+3i)	36.9	36.9	7+24	73.8
(2+i)(1+2i)	26.6	63.4	5i	90

Complex multiplication - Angles



When complex numbers multiply

The

Sum of angles of complex numbers

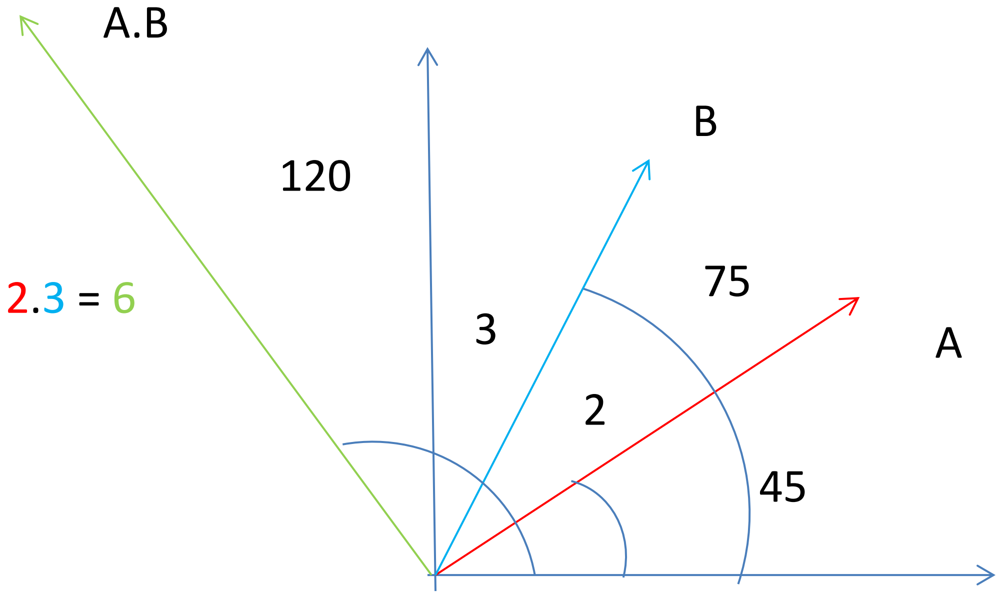
= Angle of resulting complex number

Exercise

$$r = \sqrt{x^2 + y^2} = |z| \quad \text{and} \quad \theta = \tan^{-1}(y/x)$$

Problem	Distance 1	Distance 2	Results	Results Distance
(4+3i).i	5	1	-3+4i	5
(4+3i).2i	5	2	-6.8i	10
(4+3i).(4+3i)	5	5	7+24i	25
(2+i)(1+2i)	$\sqrt{5}$	$\sqrt{5}$	5i	5

Complex multiplication -Distance



Distance from origin multiply while
Angles add

45 +75 =120

Quaternion Rotation

A Euclidean vector such as $(2, 3, 4)$ or (a_x, a_y, a_z) can be rewritten as $2\mathbf{i} + 3\mathbf{j} + 4\mathbf{k}$ or $a_x\mathbf{i} + a_y\mathbf{j} + a_z\mathbf{k}$, where **i**, **j**, **k** are unit vectors representing the three Cartesian axes.

A unit vector $\vec{u} = (u_x, u_y, u_z) = u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k}$

And its rotation through an angle of θ around the axis defined by

$$\mathbf{q} = e^{\frac{\theta}{2}(u_x\mathbf{i}+u_y\mathbf{j}+u_z\mathbf{k})} = \cos\frac{\theta}{2} + (u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k})\sin\frac{\theta}{2}$$

Quaternion Rotation

Given the unit quaternion **q** = *w* + *x***i** + *y***j** + *z***k**, the equivalent left-handed (Post-Multiplied) 3×3 rotation matrix is

$$Q = \begin{bmatrix} 1-2y^2-2z^2 & 2xy-2zw & 2xz+2yw \\ 2xy+2zw & 1-2x^2-2z^2 & 2yz-2xw \\ 2xz-2yw & 2yz+2xw & 1-2x^2-2y^2 \end{bmatrix}$$

Rotation Matrixes

Clockwise/left-hand rotation sequence with Euler angles *(ψ, ϑ, φ)*

$$
\begin{aligned}
[\,x \quad y \quad z\,] &= [\,X \quad Y \quad Z\,] R_z(\psi) R_y(\theta) R_x(\phi) \\
&= [\,X \quad Y \quad Z\,] \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \\
&= [\,X \quad Y \quad Z\,] \begin{bmatrix} \cos\theta\cos\psi & -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}
\end{aligned}
$$

Rotation to Code

```
#define PI 3.14159

GLfloat matrixX[16];
GLfloat matrixY[16];
GLfloat matrixZ[16];
GLfloat x, y, z, w;

static GLint  RotateY=0;                /* model rotation Y index*/
static GLint  RotateX=0;                /* model rotation X index*/
static GLint  RotateZ=0;                /* model rotation X index*/



}
```

Rotation to Code

```
void CreateFromAxisAngle(GLfloat X, GLfloat Y, GLfloat Z, GLfloat degree)
{

  /* First we want to convert the degrees to radians since the angle is assumed to be in radians*/
        GLfloat angle = (GLfloat)((degree / 180.0f) * PI);

 /*  Here we calculate the sin( theta / 2) once for optimization */
        GLfloat result = (GLfloat)sin( angle / 2.0f );

 /* Calcualte the w value by cos( theta / 2 ) */
        w = (GLfloat)cos( angle / 2.0f );

 /* Calculate the x, y and z of the quaternion */

        x = (GLfloat)(X * result);
        y = (GLfloat)(Y * result);
        z = (GLfloat)(Z * result);
```

Rotation to Code

```
void CreateMatrix(GLfloat *pMatrix)
{
        // First row
        pMatrix[ 0] = 1.0f - 2.0f * ( y * y + z * z );
        pMatrix[ 1] = 2.0f * (x * y + z * w);
        pMatrix[ 2] = 2.0f * (x * z - y * w);
        pMatrix[ 3] = 0.0f;

        // Second row
        pMatrix[ 4] = 2.0f * ( x * y - z * w );
        pMatrix[ 5] = 1.0f - 2.0f * ( x * x + z * z );
        pMatrix[ 6] = 2.0f * (z * y + x * w );
        pMatrix[ 7] = 0.0f;

…
…
}
```

Rotation to Code

```
void display(void){
....
CreateMatrix(matrixX);                             /* initial quatonion */
   CreateFromAxisAngle(1, 0, 0,RotateX);    /* quatonion for x rotation */
   glMultMatrixf(matrixX);                        /* multiply original matrix */


....
glPushMatrix();
      glColor3f(0.0,1.0,0.0);
      glTranslated(0.0,0.0,0.0);
   glutSolidTeapot(1.5);
   glPopMatrix();
   glutSwapBuffers();
}
```

Rotation to Code

```
void Specialkeys(int key, int x, int y)
{
   switch(key)
   {
   case GLUT_KEY_UP:
         RotateY = (RotateY +5)%360;
   break;

   case GLUT_KEY_DOWN:

         RotateZ = (RotateZ -5)%360;
   break;
…..
}
```