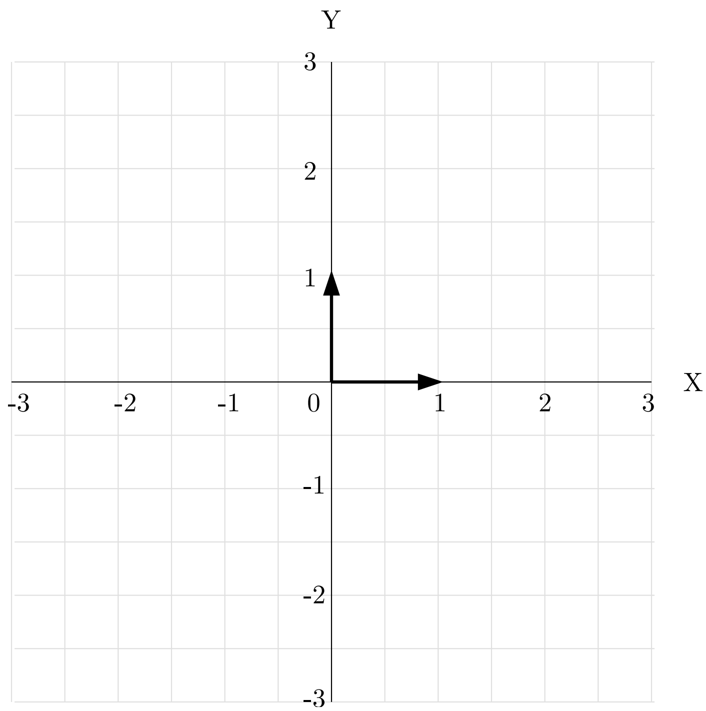
2007/2008 Graphics Tutorial 1

Before we start, we need to discuss *coordinate systems*. In 3D, they come in two flavors: *left-handed* and (you guessed it) *right-handed*. Consider the 2D-coordinate system below:



This is what we are used to: the vector $(1, 0)$, indicating the positive X-direction, points to the right, and $(0, 1)$ denotes the positive Y-direction, which is upward.

Now, if we want to extend this coordinate system to 3D, then these base vectors become $(1, 0, 0)$ and $(0, 1, 0)$, respectively. The third base vector, $(0, 0, 1)$, is orthogonal to the first two, and points towards the positive Z-direction. But what direction is that? Coming "out of the paper", or pointing "into the paper"?

Both options are possible. If we consider the XY-plane to be the "ground plane", then it makes sense to consider the Z-direction to be the vertical upward direction (i.e., $(0, 0, 1)$ comes out of the paper. On the other hand, if we consider the XY-plane to be aligned with the computer screen, then it makes sense to consider the object that are projected on the screen (while viewing a 3D model) to have positive Z-coordinates, i.e., objects that are farther away have higher Z-values. In other words, $(0, 0, 1)$ points into the paper.

A coordinate-system is left-handed if you can point with the thumb, index finger and middle finger of your left hand into the direction of the positive X-, Y-, and Z-axis, respectively, and it is right-handed otherwise.

Problem 1 *Are the two above-mentioned coordinate systems, based on the "ground plane" and the "computer screen", respectively, left- or right-handed? Is really true that aligning the thumb, index finger and middle finger with a given coordinate system is only possible with one hand, and not the other? Try it out (and watch your fellow students make funny movements with their hands).*

In graphics, we normally work with two different coordinate systems. The XYZ-system is called the "world coordinate system". This is the coordinate system in which the locations of the objects and the light sources are specified. The other coordinate system is that of the camera, and it is indicated by UVW (where U can be seen as "the X-axis of the camera, V as its Y-axis, and W as its Z-axis). The origin of the camera is the center of projection (the "view-point", or the "eye", so to speak), and we denote it with e. Its coordinates are obviously $(0, 0, 0)$ in the UVW system, but may be different in the XYZ system.

We have to decide for both systems whether we want them to be left-handed or right-handed. Last years textbook (by Slater et al.) adopted the convention of a left-handed world coordinate system, and a right handed viewing coordinate system. The book we use this year (by Shirley) chooses both systems to be right-handed, and that is what we will do in the lectures, tutorials, and programming assignments too.

To simplify things, we will initially assume that the XYZ and UVW systems coincide–the distinction vanished, and we can pretend to have only one (XYZ) coordinate system. Later, when we are going to move our virtual camera around, we will introduce the UVW system.

Problem 2 *With our choice of a right-handed coordinate system, does the vector* $(0, 0, 1)$ *points into or out of the screen if we consider the computer screen to be aligned with the XY-plane?*

Next, we consider the *viewing window*: this is a rectangle that is orthogonal to the Z-axis of the camera, and it can be considered as a window through which we view the scene (with our eye at e). The window is fully specified by five parameters: $left$, $bottom$, $right$, top, and $near$. The first four specify the $x-$ and y-coordinates of the corners of the window. The fifth parameter determines the z-coordinates of the points in the window (which is equal for all points, since the window is orthogonal to Z-axis). The name of the parameter is derived from the concepts "near plane" and "far plane", which we will discuss later in the course. Usually, the window is centered around the Z-axis, so that $left = -right$, and $bottom = -top$ (this is not required, but very common, and we will assume it for now). The value of $near$ is negative with our choice of coordinate system (i.e., the Z-axis of the camera points "out of the screen", and we are looking into the negative direction of the Z-axis). Note that this answers the problem above...

A viewing window doesn't have pixels, but a computer screen does. The number of pixels horizontally and vertically are denoted with $width$ and $height$, respectively. Note that there is no direct relation between these two parameters and the parameters of the camera described above. For instance, if we have a screen with a resolution of 640×480, it doesn't necessarily mean that $left = -320$, $right = 320$, $bottom = -240$, and $top = 240$. Yet, we need to be able to determine a mapping from pixel centers to the viewing window, in order to be able to compute rays from the origin through the pixels. Such a mapping is called a "windowing transform". Normally, the aspect ratio of the screen will be the same as that of the viewing window, but this isn't necessary either (having different aspect ratios just means that our image will be distorted). Pixels are denoted with ordered pairs (i, j), denoting the pixel in the i-th column (from the left) and the j-th row (from bottom to top). Note that we start counting from 0.

Problem 3 *Given a viewing window with parameters $left$, bottom, right, top, and near, and a screen with parameters width and height, what are the coordinates of the center of the pixel with indices (i, j)?*

In the lecture of last Thursday, we have seen that the parametric equation of a line has the form $p + tv$, where p is a point, v a vector, and t a real value. If t is restricted to $[0, \infty]$, then the equation denotes the ray in the direction of v, with origin p.

Problem 4 *Given the camera and screen specification of the previous problem, what is the parametric equation of the ray with origin e (the origin of the camera, which we assume to coincide with the origin of the world coordinate system for the time being) through the center of pixel (i, j)?*

The implicit equation of a plane in 3D has the form $ax + by + cz = d$, where $(a, b, c,)$ is a *normal vector* of the plane (if you don't know what normal vectors are, don't worry: we'll discuss them *a lot* in the near future).

Problem 5 *What is the equation of the XY-plane?*

Problem 6 *Given a plane with implicit equation $ax + by + cz = d$ and a line with parametric equation $p + tv$, what is the intersection of the plane and the line?*

The implicit equation of a sphere in 3D with center $c = (c_x, c_y, c_z)$ and radius r has the form $(x - c_x)^2 + (y - c_y)^2 + (z - c_z)^2 = r^2$.

Problem 7 *Given a sphere with implicit equation $(x - c_x)^2 + (y - c_y)^2 + (z - c_z)^2 = r^2$ and a line with parametric equation $p + tv$, what is the intersection of the sphere and the line?*

For some people, it helps to fill in concrete values in the abstract formulas in the previous two problems, to "get a feel" of what is going on. If you are one of those people, then:

Problem 8 *Define some "easy" lines, planes, and spheres, redo the previous two problems with conrete parameters, and see if the answers agree with your intuition.*