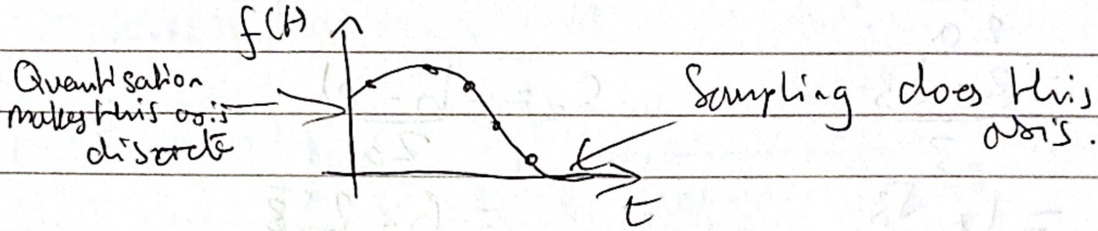
Graphics Supo 1 work

1. $i(x, y, \{0,1,2\}) = x \cdot 3 + y \cdot 3 \cdot n_{cols} + c$

2. Sampling is mapping a continous function to a discrete one. Quantisation is mapping a continuous variables to discrete ones.



3. 2D: Occlusion, relative size, shadow and foreshortening, horizon distance, focus, shading, relative brightness, familiar size.

   3D: focal depth ??

4. It would be difficult to know where to start the rays since we couldn't know where was visible, ie had no obstructions. Also reflected rays wouldn't be able to be traced.

5. Sphere eq: $P \cdot P - 1 = 0$

$a = D \cdot D$ $= 3$ $\quad b = 2D \cdot (O - C)$ $= -6$ $\quad c = O \cdot O - r^2$ $= 2$

$$S_1 = \frac{-b + d}{2a} = \frac{3 + \sqrt{3}}{3} = 1 + \frac{\sqrt{3}}{3}$$

$$d = \sqrt{36 - 24} = \sqrt{12}$$

$$S_2 = \frac{-b - d}{2a} = \frac{6 - 2\sqrt{3}}{6} = 1 - \frac{\sqrt{3}}{3}$$

6. Super-sampling. Shoot multiple rays through each pixel and average the result.

7. A pinhole camera has a singular point that all the rays shoot from. A finite aperature has multiple so it has an adjustable focal length so objects at other distances will be blurred.

8. They are good because they are always planar so it's never ambigous how they should be drawn.
They are bad since to make objects look realistic we need to draw thousands of triangles since most real objects aren't straight edged, they are curved.

9. Since transformation by a non-orthogonal matrix does not preserve angles.

Longer Qs

1. Cylinder: $(x-x_b)^2+(y-y_b)^2=r^2 \qquad z_b \leq z \leq z_b+h$

~~Ray~~: Ray: $\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} + s \begin{pmatrix} x_d \\ y_d \\ z_d \end{pmatrix}$

$$x^2-2x\cdot x_b+x_b^2+y^2-2y\cdot y_b+y_b^2=r^2$$

$$\Rightarrow (x_0+sx_d)^2-2(x_0+sx_d)x_b+x_b^2+(y_0+sy_d)^2-2(y_0+sy_d)\cdot y_b+y_b^2=r^2$$

$$x_0^2+2x_0x_ds+s^2x_d^2-2x_0x_b-2x_dx_bs+x_b^2+y_0^2+2y_0y_ds+s^2y_d^2-2y_0y_b-2y_by_ds+y_b^2=r^2$$

$$s^2(x_d^2+y_d^2)+s(2x_0x_d-2x_dx_b+2y_0y_d-2y_by_d)+x_0^2-2x_0x_b+x_b^2+y_0^2-2y_0y_b+y_b^2-r^2=0$$

$a=x_d^2+y_d^2$, $b=2x_0x_d-2x_dx_b+2y_0y_d-2y_dy_b$, $c=x_0^2+y_0^2-2x_0x_b-2y_0y_b+x_b^2+y_b^2-r^2$

$$s=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

2. Diffuse reflection is an equally scattered reflection that always occurs. It's calculated as $I_L k_d \cos\theta$ where I_L is the light intensity, k_d is the proportion of the light that's diffusely reflected and θ is the acute angle between the vector pointing towards the light source and the normal vector.

Specular reflection is the reflected light that reflects ~~at the~~ fully at an angle to the surface.

The cosine in the diffuse term is there so that light rays hitting at different angles scatter differently. The shallower the angle, the less diffuse reflection. If $\cos\theta < 0$ then the ray hit the other side of the surface so no diffuse ~~light~~ reflection.

Ambient approximates ~~the~~ all the light reflected off other surfaces onto that point.

On camera movement, diffuse and ambient will stay the same, but specular will change since the view vector changes.

3. Reflections: Recursively trace rays as they bounce off other objects. Combine this with Phong model.

Refraction: Trace rays through objects but change their direction according to the objects refractive index.

Shadows: Trace a shadow ray from intersection points to light sources and check if they intersect anything before the light. If they do then remove specular and diffuse components for that point ~~and~~ from that light.

4. To allow all the matrix transformations that we need to be ~~abstracted~~ multiplied together into one matrix.

5. The model matrix takes an object from it's own object coordinates, to the world coordinates.
The view matrix takes the objects in world coordinates and positions them relative to the camera. ~~The camera is at the origin~~
The projection matrix projects 3D coordinates to 2D screen coordinates. It keeps z coordinate for depth testing. $-1 \leq x, y \leq 1$.

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ w_s \end{bmatrix} = P \cdot V \cdot M \cdot \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$