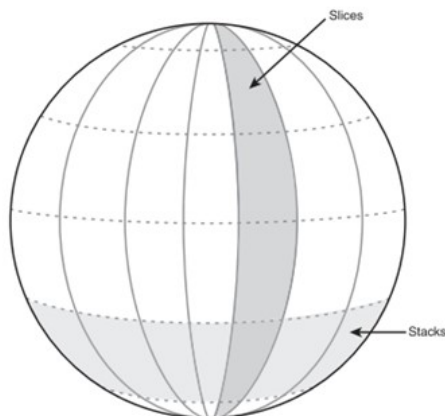


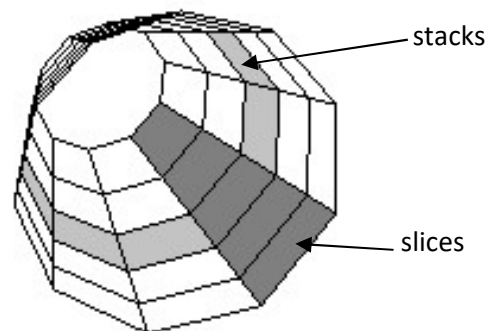
Quadrics: Rendering Sphere and Cylinder

GLU provides routines to model and render a variety of polygonal shapes, such as sphere and cylinders, which can be calculated with quadric equations.

1. Link your project to GLU.
 - a) First include the **GLU.h** header files, which also can be found under **gl** directory.
 - b) Link to GLU32.lib static library.
2. First, let's concentrate on rendering a sphere on the screen.
 - a) **GLUquadricObj *var=NULL;**
A quadrics object is stored in a **GLUquadricObj**. You have to first define a quadrics object before you can start using it.
 - b) **(GLUquadricObj *) var = gluNewQuadric();**
This function will create a new quadrics object and returns a pointer to it. A null pointer is returned if the function routine fails.
 - c) **gluSphere(GLUquadricObj *, radius, slices, stacks);**
This function will draw a sphere of the given radius, positioned at the origin. The sphere is subdivided around the z-axis into a number of slices (longitude) and stacks (latitude). The greater the slices and stacks, the smoother sphere you will have.
 - d) Compile your program, you should get a sphere on your screen. Now, Do you see a sphere? (or just a circle?).
 - e) Now, Define the quadric draw style before define the glusphere.
gluQuadricDrawStyle(quadric, GLU_LINE);
 - f) **gluDeleteQuadric(GLUquadricObj *);**
This function will destroy the target quadrics object and free up any memory used by it.
3. **gluCylinder(GLUquadric obj *, baseRadius, topRadius, height, slices, stacks);**
Change the sphere to a cylinder. study the result.



A quadric sphere's stacks and slices.



A quadric Cylinder's stacks and slices

Practical Exercise 4B

1. Draw 2 scopes of ice-cream in a wafer cone. Rotate the ice-cream in y axis to have a 3D feels. Feel free to add any toppings to the ice-cream! (7 marks)



Criteria	Marks
Complexity of models	4
Rotation	1
toppings	2
TOTAL	7