**Assignment No. 08**

8. Implement animation principles for any object

|  |
| --- |
| **Aim** |
| Animation : Implement any one of the following animation assignments |

|  |  |
| --- | --- |
| **Objective(s)** | |
| **1** | To learn different types of animation |
| **2** | To learn OpenGL function which support for animation |

|  |
| --- |
| **Theory** |
| Motion can bring the simplest of characters to life. Even simple polygonal shapes can convey  a number of human qualities when animated: identity, character, gender, mood, intention,  emotion, and so on. Very simple    A movie is a sequence of frames of still images. For video, the frame rate is typically 24 frames per second. For film, this is 30 frames per second.    In general, animation may be achieved by specifying a model with n parameters that identify  degrees of freedom that an animator may be interested in such as  • polygon vertices,  • spline control,  • joint angles,  • muscle contraction,  • camera parameters, or color.  With n parameters, this results in a vector q in n-dimensional state space. Parameters may be  varied to generate animation. A model’s motion is a trajectory through its state space or a set of motion curves for each parameter over time, i.e. ~q(t), where t is the time of the current frame.  Every animation technique reduces to specifying the state space trajectory.  The basic animation algorithm is then: for t=t1 to tend: render(~q(t)).  Modeling and animation are loosely coupled. Modeling describes control values and their  actions.  Animation describes how to vary the control values. There are a number of animation  techniques,  including the following:  • User driven animation  – Keyframing  – Motion capture  • Procedural animation  – Physical simulation  – Particle systems  – Crowd behaviors  • Data-driven animation  **Keyframing**  Keyframing is an animation technique where motion curves are interpolated through states at  times, (~q1, ..., ~qT ), called keyframes, specified by a user  **Kinematics**  Kinematics describe the properties of shape and motion independent of physical forces that  cause motion. Kinematic techniques are used often in keyframing, with an animator either setting joint parameters explicitly with forward kinematics or specifying a few key joint orientations and having the rest computed automatically with inverse kinematics.  **Forward Kinematics**  With forward kinematics, a point p is positioned by p = f(\_) where\_is a state vector (θ1,  θ2, ...θn) specifying the position, orientation, and rotation of all joints.  For the above example, p  p = (l1 cos(θ1) + l2 cos(θ1 + θ2), l1 sin(θ1) + l2 sin(θ1 +  θ2)).  **Inverse Kinematics**  With inverse kinematics, a user specifies the position of the end effector, p, and the algorithm has to evaluate the required give p. That is = f−1(p).  Usually, numerical methods are used to solve this problem, as it is often nonlinear and either  underdetermined or over determined. A system is underdetermined when there is not a unique  solution, such as when there are more equations than unknowns. A system is over determined  when it is inconsistent and has no solutions. Extra constraints are necessary to obtain unique and stable solutions. For example, constraints may be placed on the range of joint motion and the solution may be required to minimize the  kinetic energy of the system.  **Motion Capture**  In motion capture, an actor has a number of small, round markers attached to his or her body  that reflect light in frequency ranges that motion capture cameras are specifically designed to  pick up |