3451 Midterm topics for revisions

1   2D GEOMETRY

1.1   Points, vectors, dot product

Formula for dot product dot(U,V) and properties

Formula for a version R(V) of V rotated 90 degrees

Formula for V^2 and for |V|

Formula for isLeftTurn(A,B,C) testing whether polygon{..A,B.C..} turns left at B

1.2   Lines, planes, edges, rays, intersections, reflections

Test whether edge(A,B) intersects edge(C,D)

Normal projection, project(P,A,B) of point P onto line(A,B)

Test whether project(P,A,B) falls onto edge(A,B)

Parametric form of point P(t) on edge(A,B)

Implicit equation of line(A,B) through A and B

Computation of the intersection of line(A,B) with line(C,D)

1.3   Triangles

Center and area of triangle(A,B,C)

Test, pmc(P,A,B,C), returning true when point P lies in triangle(A,B,C)

1.4   Polyloops

Estimates, T and N, for the tangent and normal at vertex B in polyloop {… A,B,C…}

Displacement vector for vertex B in polyloop {… A,B,C…} during a step of the cubic Bezier subdivision

Displacement vector for midpoint of edge(B,C) in polyloop {… A,B,C,D…} during a step of the 4-point subdivision

Algorithm for performing one step of the Jarek subdivision $J_{0.5}$

Algorithm for performing one step of a polyloop smoothing

Algorithm for point-in-polygon test

Algorithm for computing the area surrounded by polyloop with vertices P[i], assuming n(i) gives the next index

1.5   Distances

Formula for distance, d(A,B) between points A and B

Math formula for the distance d(S,T) between two sets, S and T.

Provide an algorithm for computing the closest pair of a set of sites (points) P_i in the plane

Algorithm for computing d(S,T) when S is a set of points $\{S_i\}$ and T is a set of points $\{T_j\}$

Algorithm for computing the distance between edge(A,B) and edge(C,D)

Math formulae (2 versions) for the Hausdorff distance h(S,T) between two sets, S and T.

Algorithm for computing h(S,T) when S is a set of points $\{S_i\}$ and T is a set of points $\{T_j\}$

Example why we cannot compute the Hausdorff distance of two polyloops by considering all pairs of edges

Algorithm for computing the Hausdorff distance between edge(A,B) and edge(C,D)

Interpretation (love story) and applications of d(S,T) and h(S,T)

1.6   Frames

Location of point P given its coordinates (x,y,z) in frame (O,I,J,K)

Local coordinates (x,y,z) of point P in frame (O,I,J,K)

2 Animation

2.1 Collision, shock

Testing when a disk with initial center C and radius r traveling with constant velocity V will collide with line (A,B)

Algorithm for computing the velocity of the disk after the above elastic collision

2.2 Motions

Equation of the free form trajectory C(t) of the center of a ball having initial velocity V and position P

Algorithm for animating that trajectory

Algorithm for animating the bi-linear motion of a stick between edge(A,B) and edge(B,C)

Computation of a point P(t) on moving along a cubic path that starts at A with velocity U and ends at D with velocity V.

3 Computational geometry

3.1 Convexity

Test whether a polyloop $\{P_i\}$ is convex (assume n(i) and p(i) give you the next and previous index around loop)

Intuitive understanding of a convex hull of a set of points

3.2 Delaunay / Voronoi duality

Define a Delaunay triangle, given a set of sites (points) P_i in the plane

Define a Voronoi region, given a set of sites (points) P_i in the plane

Explain the duality between Delaynay triangles and Voronoi regions

Provide an algorithm for computing the point furthest from all sites P_i

4 Topology

4.1 Set operators

Math definitions of complement !S, interior iS, boundary bS, exterior eS, closure cS of a set S and examples

Math definitions of union A+B, intersection AB, and difference A–B between two sets

Recognition of sets defined by a Boolean (CSG) formula, such as A–(B–CD)

Construction of a CSG formula for a given set in 2D

Algorithm for testing whether a point P is in a region represented in CSG

4.2 Interference

Test whether disk(C_1,r_1) and disk(C_2,r_2) interfere

Test whether circle(C_1,r_1) and circle(C_2,r_2) interfere

Algorithm for testing whether two polygons (connected sets bounded by straight edges) interfere

4.3 Boundary representation of 2D regions

A polyloop is clean when it is free of self-intersections and self crossings. How would you test this?

A region R is represented by a set of clean polyloops. How to test whether a candidate point Q is in R?

A region R is represented by a set of polyloops. Semantics and algorithm to test whether a candidate point Q is in R

5 Rendering

5.1 Ray tracing in 2D

Parametric form of point P(t) on ray(V,T) from viewpoint V in the direction T

Algorithm for computing the testing whether ray(V,T) intersects with edge(A,B)

Algorithm for computing the point where of ray(V,T) intersects with edge(A,B)

Algorithm for computing the intersection of a ray with a polygon

Formula for the reflected direction, reflect(V,N), of a ray arriving with direction V on a surface with normal N

5.2 Rendering and texture mapping

Processing code to shade a polygon with vertices $\{P_i\}$

Processing code to shade and texture map a polygon with vertices $\{P_i\}$ assuming that the texture is already active. Explain what the texture coordinates mean.

6 3D geometry

6.1 Cross-product

Explain the properties of U×V

How to compute the normal to triangle(A,B,C) that is proportional to the area of the triangle

How to compute the unit normal to triangle(A,B,C)?

How to test whether triangle(A,B,C) appears clockwise to a viewer at V?

6.2 Distances and intersections

Formula for ray-plane intersection point?

Test whether ray(V,T) hits triangle(A,B,C)

Distance between a point P and edge(A,B)

Test whether point P lies inside infinite cylinder, cyl(C,D,r) with radius r and axis through points C and D

Test whether edge(A,B) intersects cyl(C,D,r)

Test whether triangle(P,Q.R) intersects cyl(C,D,r)

7 Triangle meshes

7.1 Topological properties

When is a mesh manifold?

When is a manifold mesh connected?

How many handles does a connected manifold mesh of T triangles and V vertices have?

Assume that two connected manifold meshes U and W do not interfere, how to test whether one is surrounding the other?

7.2 Orientation

How is a triangle orientation encoded?

When are two adjacent triangles of a mesh consistently oriented?

Explain the convention for orienting a connected manifold mesh

Assume a collection of connected manifold meshes representing the boundary of a solid. Explain how to orient them properly.

7.3 Corner table construction and operators

Explain the Corner Table data structure (what each table contains)

Naïve algorithm for computing O from V

Semantics and implementation of the corner operators: n(c), p(c) o(c), v(c), g(c), t(c), l(c), r(c), b(c), s(c)

7.4 Mesh processing algorithms

Algorithm for computing the valence of each vertex

Algorithm for estimating the normal N[v] of each vertex v

Algorithm for smoothing a mesh

Test whether the edge common to t(c) and t(o(c)) is convex

Algorithm for adding triangle(g(p(c)),g(n(c)),g(b)) to the mesh and restoring O

Algorithm for computing the number of shells in a manifold mesh

Algorithm for counting the number of holes in a mesh (that is a manifold with border)