CS531 Computational Geometry

Problem Set 3
Convex Hull of 3D Points

Handed out: Monday, March 26
Due: Wednesday, April 9

Problem 1. Algorithm

(a) Describe the representation of a convex polyhedron. Show the representation of a tetrahedron.
(b) Explain how to represent the conflict graph by adding pointers to this representation.
(c) Which vertices, edges, and faces are removed when a vertex is added to the hull?
(d) How can one horizon edge be found? How can the next one in clockwise order be found?
(e) Which vertices, edges, and faces are added to the hull? Which are removed?
(f) How does the conflict graph change?

Problem 2. Implementation

Implement the convex hull algorithm. The input, read from standard input, is n vertices:

```
n x1 y1 z1 x2 y2 z2 ... xn yn zn
```

and the output, written to standard output, is the t triangles of the convex hull:

```
t a1 b1 c1 a2 b2 c2 ... at bt ct
```

where a triangle is represented by the indices of its vertices in counterclockwise order when viewed from outside the hull.

Debugging

You can use the paraview visualization program to debug your program. Paraview reads a set of triangles from a file and displays it in a window that supports interactive manipulation. The attached C++ function

```
void outputVTK (const Points &pts, const vector<int> &data,
                ostream &ostr)
```

creates an input file. The vertex coordinates are stored in pts and the triangle indices are stored in data.

Problem 3. Delaunay Triangulation

Implement the planar Delaunay triangulation in Section 11.5, using your convex hull program. The input is n planar points in the format: $n, x_1, y_1, \ldots, x_n, y_n$. The output is the Delaunay triangles in the problem 2 format.