Winter 2021

COSC 3P98 Assignment 2 Computational Geometry: marking template

Q1: 2D Hull

Execution: **[25]**

- Initialization: [4] ✓
  - r (random): cmd line or menu
  - m (mouse)
  - not unique points [-1] (plus the algorithm will choke!)
- Convex hull: [15] ✓
  - either Greedy/Naive or Quickhull algorithm [10]
  - save hull edges in a data structure [2]
  - dynamic data structures [3]
- Peel: [6] ✓

Bonus: multi-peels **[10]** +2 Some clusters are crossed and same color.(see screenshot)
Should apply peels to clustered sets of points, each cluster-peel a different colour.

Style: **[8]**

- adequate comments [2] ✓
- modular code [2] ✓
- good use of data structures, global structures [2] ✓
- commands written somewhere for user (unless menus used) [1] ✓
- discretionary [1] ✓

Q1 TOTAL = 35 (base = **33**; including bonus = **43**)

Winter 2021

Q2: 2D Triangulation

Execution: **[25]**

- Initialization: [4] ✓
  - r (random): cmd line or menu
  - m (mouse)
  - not unique points [-1] (plus causes algorithm problems)
- Triangulation: [18]
  - Trisection: [10] -3 Having some crossed lines
    - Print # triangles created: [1]
  - Cleanup: [8] -7 Not working
    - Print # triangles cleaned up/restructured [1]
  - The above includes...
    - triangle polygons saved in data structure [2]
    - dynamic data structures [3]
- Lattice points [3] -1 Lattice should allow user to type N.

Bonus: 1 of… (1) Triangulated painting [10]
Should work as described. Should include image file output.
Should have 2 different colour modes as described.
(2) Colour-coded triangles: colours map to triangle areas as described [10]

Style: **[8]**

- adequate comments [2] ✓
- modular code [2] ✓
- good use of data structures, global structures [2] ✓
- commands written somewhere for user (unless menus used)[1] ✓
- discretionary [1] ✓

Q2 TOTAL = 22 (base = **33**; including bonus = **43**)

Winter 2021

Q3: 3D Convex Hull

Execution: **[25]**

- Initialization: [2]
  - K random points in 3D
    - not unique points [-1]
- Centre of gravity: [2]
- 3D Hull: [15]
  - Naive hull algorithm extended to 3D [11]
  - save hull polygons in a data structure [2]
  - dynamic data structures [2]
- Rendering in 3D: [6]
  - solid vs wireframe rendering [2]
  - spin the rendering using rotate2.c approach [1]

Bonus: 3D spherical hull peel **[10]**

Spheres of nested points; hull applied to each sphere; rendered with alpha channel.

Style: **[8]**

- adequate comments [2]
- modular code [2]
- good use of data structures, global structures [2]
- commands written somewhere for user (unless menus used)[1]
- discretionary [1]

Q3 TOTAL = _____ (base = **33**; including bonus = **43**)

TOTAL: 35 + 22 = 57 **(**base = **66**; max **86** with bonuses**)**