CPSC 314
Assignment 3 - Solutions

December 10, 2004

1. (5 points) Rasterization: Write the algorithm (pseudo-code) for integer only (Bresenham type) rasterization of the parabola $y = x^2$ for $1 \leq y \leq 25$.

   Since parabola is symmetric, only calculate points for $1 \leq x \leq 5$

   Every step we advance N or NE. Define error function: $d(x, y) = x^2 - y$
   Check sign of error at next mid-point $(x + .5, y + 1)$, if error is negative go NE otherwise go N.

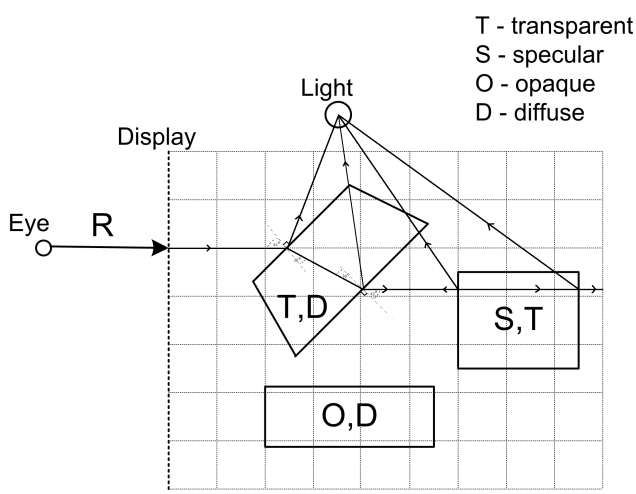
   $\Delta N = d(x + .5, y + 2) - d(x + .5, y + 1) = -1$
   $\Delta NE = d(x + 1.5, y + 2) - d(x + .5, y + 1) = 2x - 1$
   $d_0 = d(x_0 + .5, y_0 + 1) = d(1.5, 2) = \frac{1}{4}$

   Since we need integer only code, we scale initial value, ΔN and ΔNE by 4

```
x = 1
d = 1
for (y=1; y<=25; ++y) {
  plotPixel(x,y)
  plotPixel(-x,y)
  if (d < 0) {
    d += 8x - 4
    ++x
  } else {
    d += -4
  }
}
```

2. (5 points) Ray-Tracing: Draw the ray tree for the ray R shown below. Assume index of refraction c_1 for air is 1 and index of refraction for all the transparent objects in the scene is $c_2 = \frac{1}{\sqrt{2}}$. Use Snell's law to obtain refraction angles.

- Start from the eye, hit the first object
- Send a shadow ray
- object is transparent use, Snell's law.
- at exit point use Snell's law again (sending a shadow ray is implementation specific)
- Second object is specular and transparent, send reflected ray (reverse direction) and continue in object
- Send shadow ray
- Reflected ray goes all the way back to the eye, sending a shadow ray as it enters and exits the first object



3. Classify the following statements as right or wrong (provide a *short* explanation).

   (a) (1 point) If all objects in a scene have specular coefficient $k_s = (0, 0, 0)$ then using Phong model or Blin-Phong model makes no difference to the way the scene is rendered.

   True. Blin-Phong differs only by the way the specular component is calculated. In this case both algorithms have specular component 0.

   (b) (1 point) If all objects in a scene have specular coefficient $k_s = (0, 0, 0)$ then replacing a point light-source with a directional light source will not change the way the scene is rendered.

   False. Diffuse component depends on the direction of the light. With directional lights all surface points receive light from the same directions, which is not the case for point light sources.

   (c) (1 point) If a scene is illuminated with only ambient light, all the objects look flat.

   True. With ambient lighting only all points are evenly lit resulting in a silhouette which looks flat.

   (d) (1 point) A diffuse, non-specular unit cube at the origin, lit by 6 directional light sources along the $+x, -x, +y, -y, +z, -z$ directions, rendered using flat shading, looks as if it was illuminated by ambient light.

   True. Since lights are directional, each point on the surface will receive light from exactly on source which is perpendicular to the face. Therefore, all points are equally lit which is equivalent to ambient lighting.

   (e) (1 point) The polygon scan conversion (edge-walking) algorithm works for non-simple (self-intersecting) polygons.

   True. The algorithm finds the edge points on each scan-line, and fills all pixels between pairs of points. Since every intersection changes state from outside to inside the polygon or the other way around, and due to the fact that the scan-line starts outside the polygon, the algorithm is correct.

4. (5 points) Curve Continuity:

   Given the following parametric curves:

$$
\begin{aligned}
f^1(u) &= (u, u), \quad u \in [0, 0.5] \\
f^2(u) &= (1-u, 1-u), \quad u \in [0.5, 1] \\
f^3(u) &= (2u^2 - u + 0.5, 2u^2 - u + 0.5), \quad u \in [0.5, 1] \\
f^4(u) &= (2u^2, 2u^2), \quad u \in [0.5, 1]
\end{aligned}
$$

   will the curves defined in the table below be continuous (C^0, C^1, G^1) at $u = 0.5$ (each curve is defined on $[0, 1]$ by combining the functions on $[0, 0.5]$ and $[0.5, 1]$).

	C^0	C^1	G^1
f^1 and f^2	yes	no $1 \neq -1$	no
f^1 and f^3	yes	yes $4 \cdot .5 - 1 = 1$	yes (C^1 is also G^1)
f^1 and f^4	yes	no $1 \neq 4 \cdot .5$	yes, $f'^1(.5) = C \cdot f'^4(.5)$

5. **(5 points) Bonus** Given points $P_1, \ldots, P_n, P_{n+1} = P_1$ in 3D develop the formula for a smooth C^2 curve passing through them (Hint: use Hermite curves for each segment P_i, P_{i+1}).

   (5 points) n points define n segments,

$$
s_i(t) = \begin{pmatrix} x_i(t) \\ y_i(t) \\ z_i(t) \end{pmatrix} \quad i = 1, \ldots, n \quad t \in [0, 1]
$$

$$
s.t. \quad s_i(0) = P_i \quad and \quad s_i(1) = p_{i+1}
$$

   Let $x_i(t) = a_i^x t^3 + b_i^x t^2 + c_i^x t + d_i^x$ and define $y_i(t)$ and $z_i(t)$ in the same manner.
   $x_i(0) = p^i \quad x_i(1) = p^{i+1} \quad i = 1, \ldots, n \quad 2n$ equations
   $x_i'(1) = x_{i+1}'(0) \quad i = 1, \ldots, n \quad n$ equations
   $x_i''(1) = x_{i+1}''(0) \quad i = 1, \ldots, n \quad n$ equations

   So we have a linear system with $4n$ equations with $4n$ variables. Construct and solve similar systems for $y(t)$ and $z(t)$ to get a smooth curve through all the points.