This exam consists of 4 questions. Do all of your work on these pages (use the back for scratch space), giving the answer in the space provided. This is a closed-book exam, but you may use one page of notes during the exam. **Put your NetID on every page, and write out and sign the Honor Code pledge before turning in the test:**

“I pledge my honor that I have not violated the Honor Code during this examination.”

<table>
<thead>
<tr>
<th>Question</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
</tr>
</tbody>
</table>
1. Rendering roundup (18 points)

Which of the following rendering methods:

- RT: Basic Ray Tracing
- PT: Monte Carlo Path Tracing
- Rad: Radiosity
- Ras: Rasterization as in OpenGL

have each of the following properties? Choose one or more correct answers for each.

- Usually fastest for scenes with low to moderate polygon count:
- Fastest for a model with a huge number of polygons, but covering a tiny part of the screen:
- Can simulate mirror reflection on arbitrary scene objects:
- Can accurately render surfaces not illuminated by direct light:
- Can render partially-transparent materials (with or without refraction):
- Can simulate refraction:
2. **Illumination** (22 points)

Consider a ray tracer that implements the (local) illumination model with which we have been working:

\[ I_E + K_A I_A + K_D (N \cdot L) I_L + K_S (V \cdot R)^n I_L \]

For each quantity in this equation, state whether its value depends on:

- **G**: Surface geometry (shape)
- **M**: Surface material
- **L**: Property (including position) of a specific light source
- **I**: Approximation of indirect illumination in scene
- **D**: Ray direction
- **C**: Coordinates of ray/surface intersection
- **S**: Result of casting one or more new rays

Choose *one or more* correct answers for each.

- **\( I_E \)**:
- **\( K_A \)**:
- **\( I_A \)**:
- **\( K_D \)**:
- **\( N \)**:
- **\( L \)**:
- **\( I_L \)**:
- **\( K_S \)**:
- **\( V \)**:
- **\( R \)**:
- **\( n \)**:
3. **Refraction** (30 points)

Consider a recursive sequence of rays being traced through a glass object. (Index of refraction is $\eta_{\text{glass}} = 1.5$ in glass and $\eta_{\text{air}} = 1$ in air.) The original ray began at point $P_0$ with direction $D_0$, and hit the surface at $P_1$.

![Diagram of ray refraction through a glass object](image)

a) What is the direction $D_1$ of the secondary transmissive (refracted) ray? Show the derivation from Snell’s Law: $\eta_1 \sin \theta_1 = \eta_2 \sin \theta_2$. 
b) How would you test whether a light source at position $P_L$ should be used in the lighting equation at $P_1$? (Give the origin and direction of any additional rays you cast, and/or the expression for any back-facing tests you do.)

c) How would your answers to questions (a) and (b) be different for the intersection found at $P_2$? (Hint: it is not always sufficient to change indices on the previous equations.)
4. Dynamics (30 points)

Consider simulating a 1-dimensional mass-spring system, with a particle of mass 1 attached to a frictionless spring with rest length 0 and spring constant 1. The other end of the spring is fixed at $x = 0$. The initial conditions are:

- Initial position $x_0 = 1$
- Initial velocity $v_0 = 0$

a) Write down the forces on the particle, as well as the position and velocity after each of three iterations of Euler’s method with a time step of $\Delta t = \frac{1}{2}$. Please write all results as fractions rather than decimals.

b) What is the total energy in the system at each time step? Recall that kinetic energy $KE = \frac{1}{2}mv^2$ while a spring’s potential energy $PE = \frac{1}{2}k(l - l_0)^2$. 


c) Now simulate three iterations of a different method, in which the new, updated velocity is used to compute the position at each timestep, rather than the velocity from the previous timestep. (This is a special case of the “leapfrog” method for solving ODEs.) Comment on the expected stability of the two solution methods for simulating mass-spring systems.