

# Praktisches Arbeitsblatt

Armas Scharpegge

DPG Schülertagung 2025

Öffnet <https://www.shadertoy.com/view/wf1BzS> und macht euch etwas mit der Oberfläche vertraut. Ihr müsst nur Code in `Buffer A` modifizieren und dort nur `intersectScene` und `traceRay`.

1. Modifizierte `traceRay` so, dass das Bild
  - (a) einfach rot ist.
  - (b) als  $(r, g, b)$ -Wert die  $(x, y, z)$  Koordinate der Strahlungsrichtungen  $\mathbf{d} = \mathbf{ray}.d$  anzeigt.
2. Implementiere die Methode `intersectScene`. Verwendet dabei `intersectSphere` und `intersectPlane` um den global nächsten Schnittpunkt zu finden. Ihr könnt als Zwischenschritt erstmal nur die Schnittpunkte mit den `planes` bestimmen.
3. Implementiere den Fall `MATERIAL_LIGHT` für einen lambertschen Strahler, d.h. die radiance ist direkt der gespeicherte Materialwert.
4. Implementiere den Fall `MATERIAL_DIFFUSE` für eine lambertsche Oberfläche. Um die radiance entlang eines neuen Strahls, der von der Oberfläche aus in eine zufällige Richtung zeigt, zu bestimmen, musst du `nextRay` setzen. `nextWeight` gibt dabei den Vorfaktor an (d.h.  $\text{nextWeight} = \frac{f(\mathbf{p}, \omega_o, \omega_i) |\cos \theta|}{p(x)}$ ).  $\cos \theta$  kann durch das Skalarprodukt  $\mathbf{d} \cdot \mathbf{n}$  berechnet werden. `randHemisphere(n)` gibt einen zufälligen Vektor auf der Halbkugel in Richtung der Normalen  $\mathbf{n}$  zurück. Der Ursprung  $\mathbf{o}$  des Strahls sollte leicht in Richtung der Normalen verschoben werden, damit als Schnittpunkt nicht das gleiche Objekt gefunden wird. Ein Offset von  $1e-4 * \mathbf{n}$  funktioniert hier gut.
5. Implementiere einen perfekten Spiegel für `MATERIAL_SPECULAR`.
6. Welches große Problem ergibt sich bei dieser Implementation, wenn die Lichtquelle deutlich verkleinert wird? Wie könnte man es lösen?