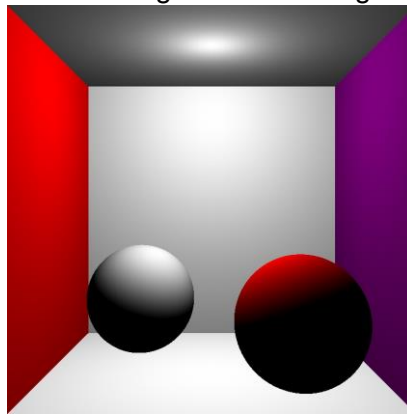


Projet de lancer de rayon - Phase 2 (25 pts)

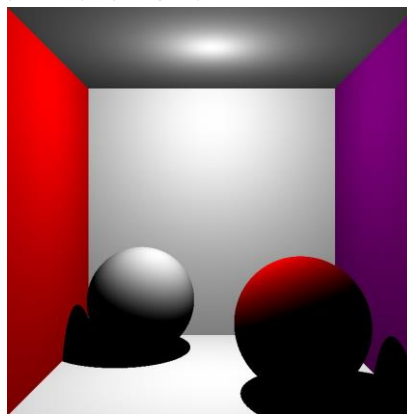
Donnons maintenant une apparence 3D aux objets

1. Mettre à jour `castRay()` avec le modèle de Phong (10 pts)

- La fonction `intersect()` doit calculer et retourner le point d'intersection et la normale en ce point en coordonnées monde
- Calculer l'illumination de Phong du point d'intersection le plus proche dans `castRay()`
 - Les paramètres de Phong sont dans `ShadingValues` pour chaque objet
 - Vous pouvez vous inspirer de `drawObject()` pour voir comment ceux-ci sont utilisés dans le shader de Phong dans le fragment shader.
 - Vous devrez utiliser `lightPosition` et `lightColor` également.



2. Mettre à jour `castRay()` pour ajouter des ombres : si l'objet intersecté est dans l'ombre il ne doit pas être illuminé (black). (15 pts)



- Les problèmes éventuels :
 - Si vous n'ignorez pas les petites valeurs de `t` dans votre code d'intersection, le calculateur d'ombre trouvera souvent une intersection du rayon avec l'objet à partir duquel il est lancé.
 - Le calculateur d'ombre ne teste pas l'intersection avec tous les objets mais si le rayon intersecte UN objet (peu importe lequel) avant d'arriver à la source lumineuse.

- Pour les ombres douces : une fois le calcul d'ombre fonctionnel, supposez que la source lumineuse est un carré lumineux (area light). Echantillonnez aléatoirement ce carré (créant ainsi plein de lumières ponctuelles) et lancez un rayon d'ombre par échantillon. Calculez le pourcentage de rayon retournant vrai et utilisé le pour atténuer la couleur finale.