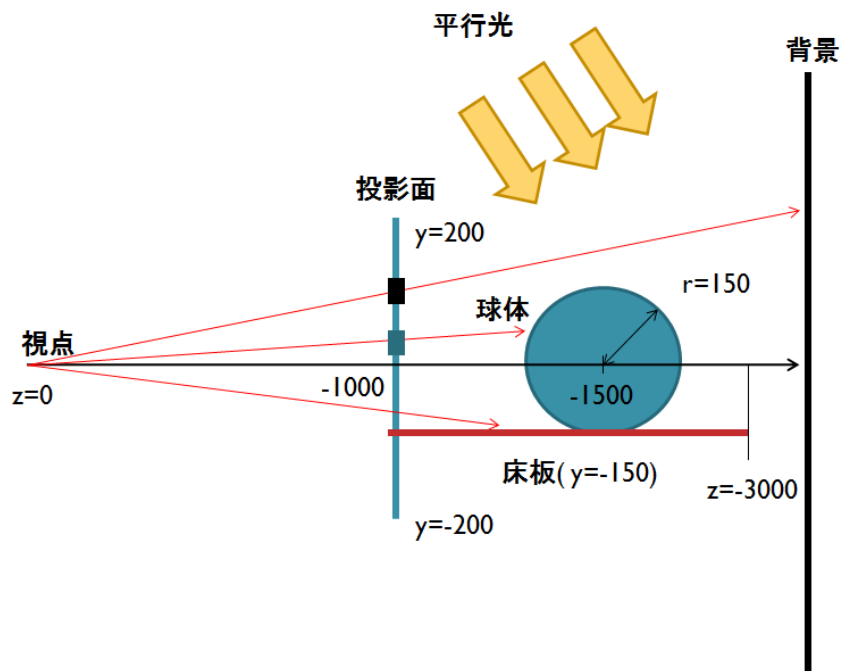


課題の目標

- ・レイトレーシングでの反射と影の扱いの原理を理解する
- ・オーバーサンプリングによるアンチエイリアシングの仕組みを理解する

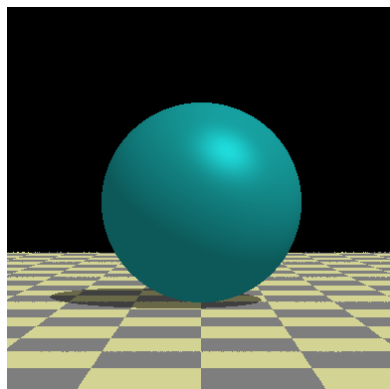
課題の内容

下図のようなシーンをレイトレーシングによってレンダリングする（床板以外は前回と同じ）。

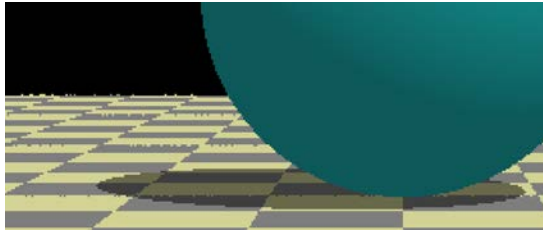


- (1) 球と床板をレンダリングしなさい。（拡散反射定数  $K_d$  の値は球も床も同じとする）
- (2) 床に格子模様を付加しなさい。格子のサイズは  $100 \times 100$  とする。
- (3) 床に球体の影が表示されるようにしなさい。影となる場所は、球体によって光線が遮蔽される場所である。影となる地点の色の計算は、簡易的に RGB の値を  $0.5$  倍することで実現する。

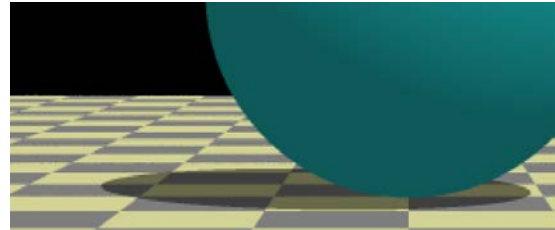
(1)～(3)の実装で、次のような結果が得られる。



- (4) 境界を滑らかに表現するアンチエイリアシングを行うことを考える。1 ピクセルを **3x3** に **9** 分割してサンプリングを行い (1 ピクセルあたり **9** 回のレイトレーシングを行う、つまり対象のピクセルから **x** と **y** 方向にそれぞれ **1/3**、**2/3** ずつずらしてレイを飛ばす)、その **9** つの値の平均値でピクセルの色を決定しなさい。



アンチエイリアシング前



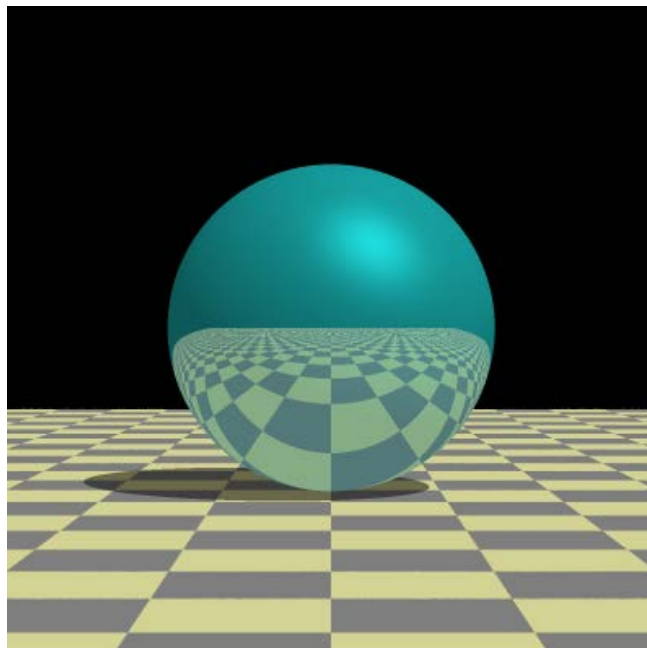
アンチエイリアシング後

#### 【発展課題】

- (1) 自由にプログラムコードを変更してサンプル以外の結果を出力する。

例：床の模様を変えてみる。背景に模様をつけてみる。球の数を増やしてみる。

- (2) 飛ばしたレイを球面で反射させ、鏡面反射を実現してみる。これには、反射させたレイが床に当たる場合に、その場所の床の色と球の色を足して **2** で割ることを行う。(複数の球体間での光の反射を扱うのは、レイを再帰的に追跡するので難易度が高い)



鏡面反射を実装し、異なる視点から見た例