

## CG課題説明書

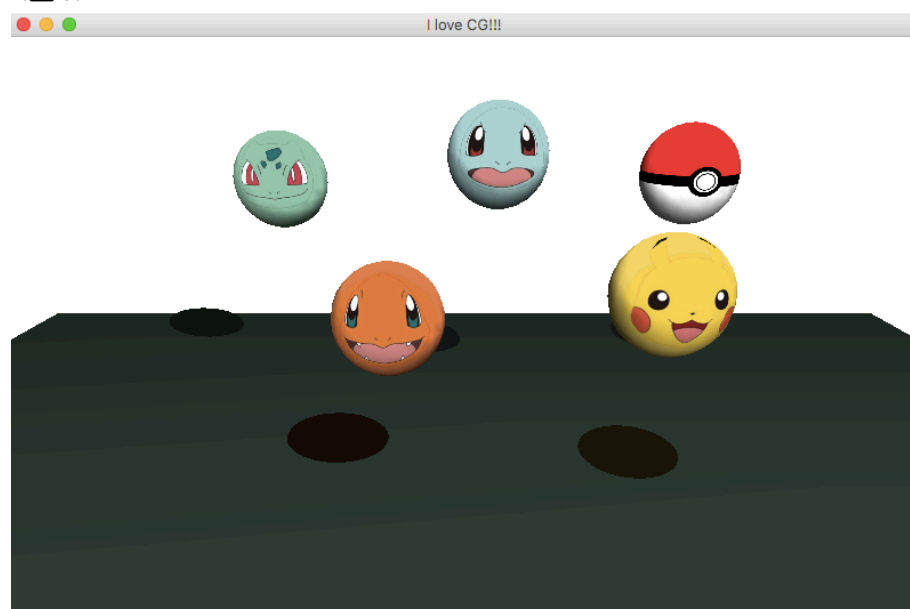
### ■ 概要

ポケモンが好きなので何かポケモンを使ったCGが見たいと思い、ポケモンがマッピングされたボールが跳ねるCGを作成した。ただ跳ねるだけではなく、時間が経つと停止するアニメーションにして、ボールの影が床につくようにした。

### ■ 操作方法

\$ ./jogl.sh を実行後は<図1>のような画面が現れる。

<図1>



画面をクリック	アニメーションをスタート
Rボタン	ボールを最初の位置に戻す
Sボタン	ボールを停止させる
Qボタン	終了

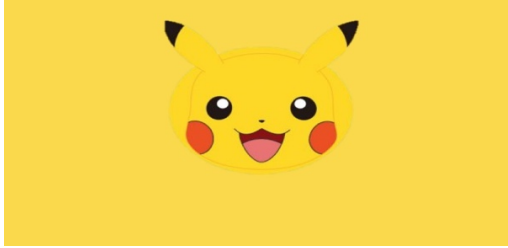
### ■ プログラム内容

新しくMyBallクラスとMyGroundクラスを作り、それぞれボールと地面の描画に用いた。

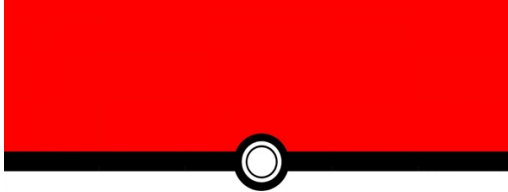
Myballクラスに関して

1. 反発係数を考慮したボールの自由落下  
スーパーボールのような跳ね方にしたかったので反発係数を0.8に設定してバウンドするたびに速度が小さくなるようにした。

2. 球体へのテクスチャマッピング  
マッピングには横長の画像を用意した。



下のような画像を使うとmonsterボールができる。



[こちらのページ](#)を参考に実装した。javaのバージョンが変わった影響で、importする場所が変わっているのものでそのままでは実行できなかった。流れとしては、画像を読み込んでオブジェクト化した後に、球体を描画するためのgluSphereメソッドの引数にオブジェクト化したものを指定した。

### 3. 影の描画

影行列を使う方法を用いた。[Østfold University College](#)さんと[OpenGLプログラミングメモ](#)さんのページがとてもわかりやすく、参考にした。

影行列を計算したあとにglMultMatrixfメソッドの引数に影行列を渡して球体へ行列の適用を行なった。

影が床につかずにボールにくっついて移動してしまったり、ボールが停止しているときの影の位置や形が明らかに違うなど様々な問題が起こり影の実装が一番大変だった。

#### ■ 参考にしたサイト

[http://www.t3japan.gr.jp/pdf2011/07\\_kataoka.pdf](http://www.t3japan.gr.jp/pdf2011/07_kataoka.pdf)  
<https://tech.nikkeibp.co.jp/it/article/COLUMN/20060818/245889/>  
[https://atelier-yoka.com/dev\\_android/p\\_main.php?file=apigluperspectiv](https://atelier-yoka.com/dev_android/p_main.php?file=apigluperspectiv)  
<http://seiya-kumada.blogspot.com/2016/04/opengl.html>  
<http://ynomura.dip.jp/archives/2013/09/opengl.html>  
<http://www.natural-science.or.jp/article/20091114182234.php>  
<https://www21.atwiki.jp/opengl/pages/182.html>  
<http://www.it.hiof.no/~borres/j3d/JOGL/simpleshadow/p-shadow.html>  
<http://www.land-of-kain.de/docs/jogl/>