塊魄(仮)



・対応ハード

　PC Windows10

・製作人数

　１人

・製作期間

　2019年５月～

・製作環境

エンジン

　 学校内製のエンジン(DirectX11)

ツール

　　 MicroSoft VisualStudio2019

　3DSMAX2020

　FireAlpaca

　GitHub

　 使用言語

　　 C++　Python3.7

・モノのデータ管理

　・参照

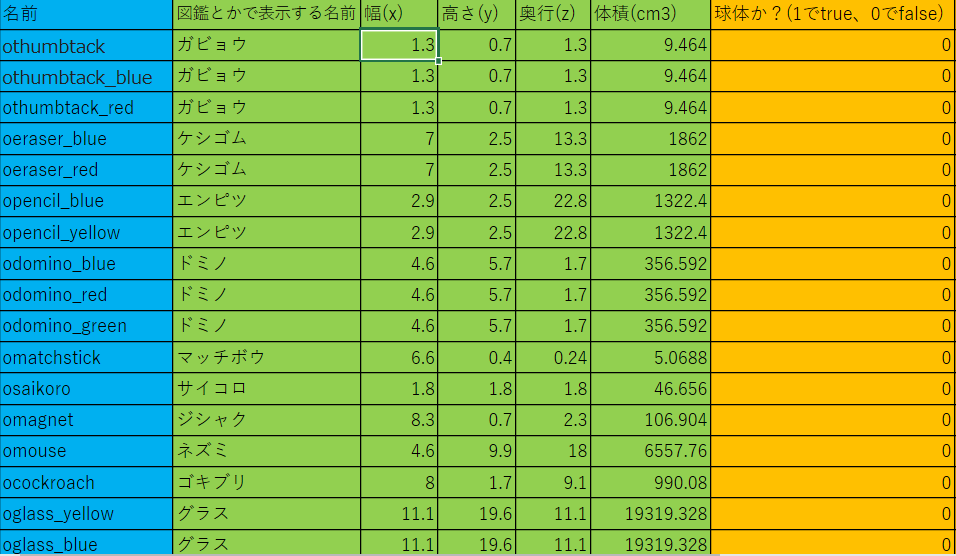
PythonApplication1 Game/Object/ObjectData.h,cpp

ゲーム内には様々なモノがありますが、それらのデータはExcelで

管理しています。そのデータをPythonでテキストドキュメントに

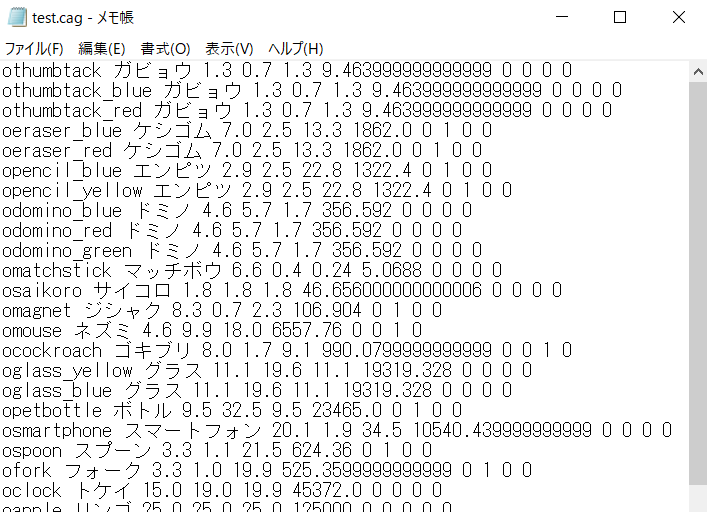
変換して、そのテキストドキュメントをゲームで読み込んでいます。

Excel



Python

テキストドキュメント



ゲームで読み込む

・オフスクリーンレンダリング

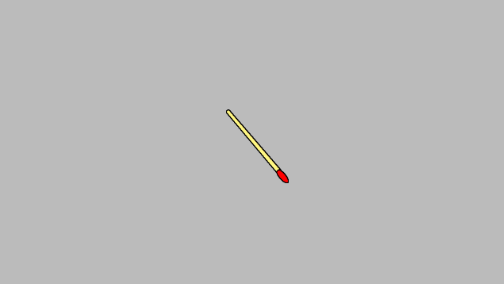
　・参照

Game/OffScreen.h .cpp Game/Object/Collection.h .cpp

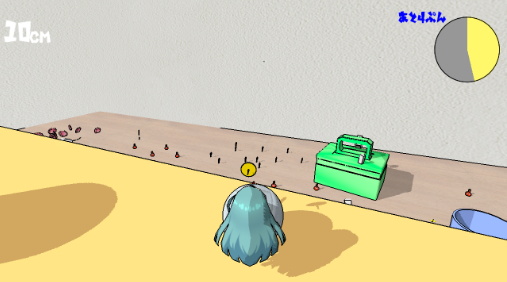
ゲーム中に右下に巻き込んだものを表示していますが、これはフレ

－ムバッファとは別のレンダリングターゲットにモデルを描画した

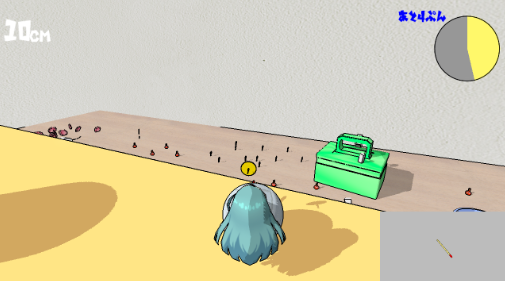
のを、フレームバッファに合成しています。

 フレームバッファ 　　オフスクリーン

テクスチャとして合成



　　　　　　フレームバッファ

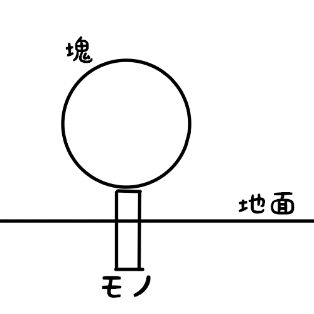


・あるオブジェクトを巻き込むと、

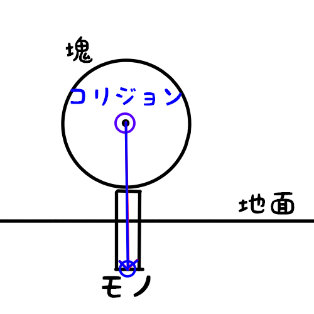
　オブジェクトの形に合わせて塊が動く

　・参照

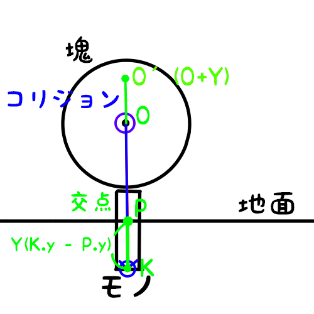
Game/LineSegment.h .cpp

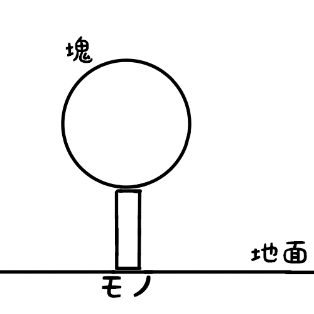


モノがこのように塊にくっついている場合、



コリジョンを塊の中央からモノの先端まで移動させます。





この時コリジョンが地面と衝突したら、モノの先端から交点までの

高さを求めて、その高さを塊の座標に加算します。

・塊にくっついた時のモノの挙動

　・参照

Game/Object/Obj.h .cpp

スキニングとよく似ているのですが、オブジェクトが塊にくっついた時、まず1フレーム目に

**モノのワールド行列×塊の逆ワールド行列**

**＝塊座標系のモノのローカル行列**

　をして塊を中心としたモノのローカル行列を求めます。

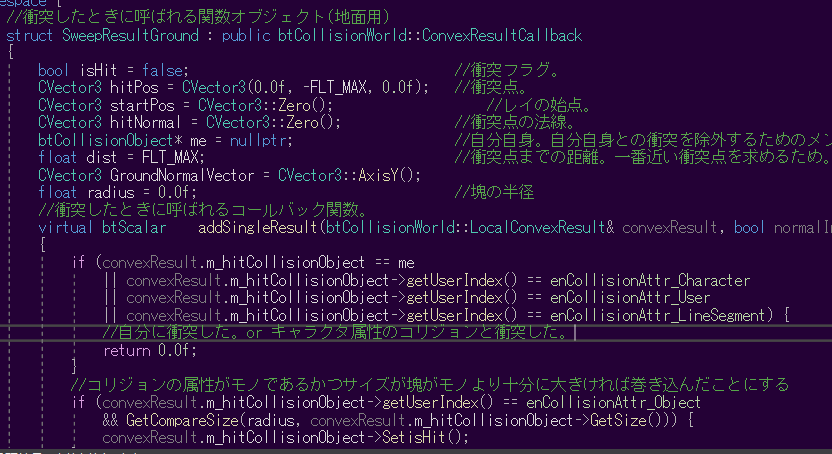
　そして毎フレーム

**塊座標系のモノのローカル行列×塊のワールド行列**

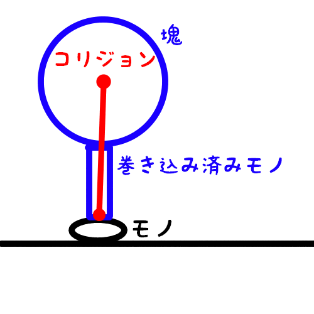
**＝モノのワールド行列**

をしてモノのワールド行列を求めています。

・塊とモノの巻き込み判定



これが塊のコリジョンが他のコリジョンと衝突した際のコールバック関数の一部なのですが、モノのコリジョンと衝突した場合、塊のサイズがモノより十分に大きければ、巻き込んだことにしています。



またこのように尖っているモノを巻き込んでいる場合には、ガタガタ処理のためにコリジョンをモノの先端まで移動させるのですが、その際も同様の処理をしています。これにより、尖ったモノを巻き込んでいてもモノを巻き込みやすくしています。

以下Qiitaで私が書いた記事のURLです。

・エッジ検出

　・参照

kgEngine/graphics/normal/EdgeDetection.h .cpp

Game/Assets/shader/edgedelection.fx

<https://qiita.com/akurobit/items/b5231ffae738810b63e7>

・円形ゲージ

・参照

kgEngine/graphics/2D/Sprite.h .cpp

Game/Assets/shader/sprite.fx

<https://qiita.com/akurobit/items/ab90b88088678f706e3a>

・トゥーンシェーダー

・参照

Game/Assets/shader/model.fx

<https://qiita.com/akurobit/items/a016ef4a022eed15c268>

・被写体が全部移るためのカメラ設定

・参照

Game/Collection.cpp

<https://qiita.com/akurobit/items/a6dd03baef6c05d7eae8>