

ボール転がしゲーム

ルール

時間内に左上にあるボールを右下まで運べたらゴールです。
ただし、落とし穴の部分に当たってしまうと初めからやり直しになってしまいます。

細かい仕様

OrbitControlsを使うことでカメラ位置をマウスのドラッグでできるようにし、その座標値を得ることでボールを動かせるようにしました。(実際には、物体をドラッグによって動かしているように見えることを利用しました。)

そのため、物体の中でボールの位置だけが動いています。

ボールは2次元で動きますが、速さや位置の計算は各軸で行うようにし、壁に当たったときには、当たった壁がある方の速さが0となるようにしました。(反発係数を用いて計算をしてもよかったのですが、今回は簡単のために速さを0にすることで、時間の初期化を行いやすくしました。)

計算の回数を減らすために、壁は摩擦がないものとし、底の部分には摩擦があるものとして計算されています。

ボールが次に移動する位置が壁を超えてしまうときに、位置を壁の手前、速度を0に設定しているため、ときどきボールの位置が変わってしまうことがあります。(どうやって壁の判定をするかを考えるのが難しかったです。)

参考にしたサイト:

OrbitControls.jsのGithub

<https://github.com/mrdoob/three.js/blob/dev/examples/js/controls/OrbitControls.js>

OrbitControl.js

<https://threejs.org/docs/index.html#examples/en/controls/OrbitControls>

参考にしたコード

<https://www.i-ryo.com/entry/2020/05/02/215245>

視点を動かす

<https://www.i-ryo.com/entry/2020/05/02/215245#作ったもの-1>

物体の回転

<https://gupuru.hatenablog.jp/entry/2013/11/27/214929>