

概要

- ・課題 1、2、3 があります。
- ・ポインタや関数・クラスといった概念、k2Engine への理解を深めることが目的です。
- ・課題提出時に、上記の内容が理解できているか簡単な質疑応答を行います。
- ・なるべく自身の力のみで課題を進めてください。
- ・課題クリア者には、ゲーム大賞のチームに参加していただきます。

課題 1. リファクタリング

ゲームの挙動は一切変更せずに、プログラムを下記の通りに修正してください。

- ・Game プロジェクトに存在するクラスのメンバ変数を全て **private** にする(メンバ関数は **public** 可)

→参考：[C++ public private アクセス指定子の使い方 | プログラミングランド \(skpme.com\)](http://skpme.com)

- ・FindGO 及び FindGOs の使用禁止

→既に FindGO 及び FindGOs を使用している箇所は削除して、別の方法で実装する

- ・Player クラスのメンバ変数 **playerState** の型を列挙型(enum)にして使用する

→参考：[C++での enum\(列挙型\)の使い方とは？ class 指定方法や文字列変換方法を紹介！ | .NET コラム \(fenet.jp\)](http://fenet.jp)

課題 2. ゲーム要素の追加

課題 1 の条件を引き継ぎつつ、ゲームに以下の要素を追加してください。

Game/Game.exe で完成品が起動します。

- ・プレイヤーを追従するエネミーを追加

- 新しく Enemy クラスを作成する

- エネミーは 3 体配置

- プレイヤー等と同じように、ゲームクリア画面等に遷移する際に全て削除する

- モデルは Assets/modelData/enemy/enemy.tkm を使用

- ・ゲームオーバーの追加

- プレイヤーがエネミーに衝突すると、プレイヤーがダウンアニメーション (Assets/animdata/KneelDown.tka) を再生した後にゲームオーバー画面に遷移

- ゲームクリアと同じように、ゲームオーバーに遷移する際にプレイヤー等が削除する

- 画像は Assets/sprite/gameover.dds を使用

- ・プレイヤーが☆を取得するとエフェクトが発生

- エフェクトは Assets/effect/get.efk を使用

課題 3. ベクトル

ベクトル $A(100,200,-100)$ 、ベクトル $B(-500,600,0)$ について、**k2Engine の Vector3 クラス**を使用して下記の値を求めなさい。

- ① ベクトル A とベクトル B の内積
- ② ベクトル A とベクトル B の成す角 θ (①の値を用いること)
- ③ ベクトル A とベクトル B の外積
- ④ ③は、ベクトル A、B にとってどのようなベクトルでしょうか。