

提出日 2025 年 01 月 28 日

# 応用プログラミング自由課題レポート

学籍番号 202211905  
所属 工学システム学類  
学年 3 年  
氏名 鈴木暁満

## 1 はじめに

これまで「2D Micro Game」をもとにしてスーパーマリオのような横スクロールアクションゲームを作成してきた。しかし、このチュートリアルでは 2D ゲームがすでに完成されており、その上でプレイヤーや敵の動きを変更することができる程度であった。面白くなかったので題材を変更し、縦スクロール型の 2D シューティングゲームを作成することにした。

## 2 参考プログラム

縦スクロール型 2D シューティングゲームを作成しているインターネット記事[1]を参考にする。

### 2.1 = 参考プログラムの基本機能

この参考プログラムは、実際に 2D ゲームを作成しながら Unity の基本的な機能を学ぶことができる。このゲームは、ゲームオブジェクト（隕石）を全て打ち落とすゲームである。具体的には、以下の機能が実装されている。

- キーボード入力によるプレイヤーの移動・アクション
- 隕石の生成、コライダーの設定による当たり判定
- 隕石の破壊
- テキストによるスコア表示

## 3 課題設定

参考プログラムで実装されていない機能の実装を目指す。具体的には、以下の機能を追加する。

- 隕石とプレイヤーの衝突によるゲームオーバー
  - 隕石が撃ち落とされなくてもゲームオーバーにはならない
- アイテムの追加とそれによるプレイヤーの強化
- スコアに対応した難易度の調整
- ゲームオーバー時のスコア表示とリスタート機能
  - リスタートとゲーム終了を選択できるようにする
- ゲームクリアの実装
- ゲームスタート画面の実装
- 音楽の追加

また、余裕があれば縦スクロールを横スクロールに変更することを目指したい。加えて、各ゲームオブジェクトのテクスチャの変更も行う。

## 4 計画概要

上記の課題設定に基づいて、以下の計画を立てる。 $\alpha$ 版で実装する機能を元として $\beta$ 版の作成を行う。使い回しのできる部分は使い回すことで、効率的に作業を進める。

### 1. $\alpha$ 版の実装

- ・ 隕石とプレイヤーの衝突によるゲームオーバー
- ・ スコアに対応した難易度の調整
- ・ ゲームオーバー時のスコア表示とリスタート機能
- ・ アイテムの追加とそれによるプレイヤーの強化

### 2. $\beta$ 版の実装

- ・ ゲームクリアの実装
- ・ ゲーム開始画面の作成
- ・ 音楽の追加
- ・ 各ゲームオブジェクトのテクスチャの変更

## 5 $\alpha$ 版

### 5.1 実装内容

以下の機能を実装した。

#### 隕石とプレイヤーの衝突によるゲームオーバー

プレイヤーにコライダーを設定し、隕石との衝突を検知することでゲームオーバーとなるようにした。サンプルでは、弾と隕石の衝突による隕石の破壊を実装していたので、それを参考にした。また、サンプルでは、隕石が落ちるとゲームオーバーであったのを、隕石がプレイヤーに当たるとゲームオーバーになるように変更した。

#### スコアに対応した難易度の調整

スコアに応じて隕石の落下速度を変更することで難易度を調整した。スコアが高いほど隕石の落下速度が速くなるようにした。

#### ゲームオーバー時のスコア表示とリスタート機能

ゲームオーバー時にリスタートとゲーム終了の選択が光るボタンを実装した。“GameOverManager.cs”を新しく作成し、ゲームオーバー時に呼び出される関数を設定した。また、unityのUI機能を使用して、ボタンを選択してリスタートとゲーム終了を選択できるようにした。

#### アイテムの追加とそれによるプレイヤーの強化

ランダムで3種類のアイテムが出現し、アイテムに応じてプレイヤーが強化されたりスコアが増加するようにした。隕石生成のスク립トを参考にした。

## 5.2 評価

難易度の調整とリスタート機能を実装したことで、ゲームの楽しさが増したと感じた。特に、スコアに応じて難易度が変化することで、プレイヤーがより高得点を目指すような仕組みができた。また、リスタート機能を実装することで、ゲームオーバー時にすぐに再挑戦できるようになり、プレイヤーのストレスが軽減されたと感じた。

### 改善点

ゲームオブジェクトの速度を計算する。現状デフォルトの速度で設定しているが全体的に速く、難易度が高くなっている。また、ゲーム性を高めるためゲーム開始画面とゲームクリア画面も作りたい。音楽も追加したい。

## 6 $\beta$ 版の実装

### 6.1 実装内容

$\alpha$ 版で実装した機能に加え、以下の機能を実装した。

#### ゲームオブジェクトの速度計算

各ゲームオブジェクトの速度を計算し、プレイに最適な速度に調整した。また、スコアに応じて速度が変化するようにした。

#### ゲーム開始画面の作成

ゲーム開始画面を作成した。ゲームスタートボタン、ゲーム終了ボタンを配置し、ゲームスタートボタンを押すとゲームが開始されるようにした。これの作成にあたり、SceneController.cs を作成した。

#### ゲームクリアの実装

スコアが一定値を超えるとゲームクリアとなるようにした。ゲームクリア時には、ゲームクリア画面が表示されるようにし、ゲームクリア画面には、スコア表示とリスタートボタンが表示されるようにした。これの実装は、ゲームオーバー時の実装を参考にした。

#### 音楽の追加

各画面とロケットの動作に合わせた音楽を追加した。

#### 各ゲームオブジェクトのテクスチャの変更

各ゲームオブジェクトのテクスチャを変更し、ゲームの見た目を変更した。

- ・隕石→エイリアン・UFO
- ・ロケット：見た目の変更
- ・アイテム、ボタン→自作のテクスチャに変更

### 6.2 評価

ゲーム開始画面と音楽を追加することで、ゲームの完成度が増した。やはり、ゲーム画面だけでなくそれに伴うゲームの要素が大切であると感じる。

### 6.3 全体の考察

作成するゲームのジャンルを変更したため作業時間が少なかったが、ゲームに求められる最低限の機能を実装することができた。また、ゲームの完成度を高めるためには、ゲーム画面だけでなく、ゲームに関連する要素も重要であることがわかった。残された課題として、未実装である ManualScene の作成が挙げられる。ManualScene は、ゲームの操作方法やルールを説明する画面である。これを作成することで、プレイヤーがゲームをより楽しむことができると思われる。また、ボタン操作時の音楽がないのは寂しい。ファミコンゲームのよ

うな電子音を追加した。加えて、ゲームのステージ設定を行うこともできるとより楽しめると考えられる。ステージに応じて、アイテムや敵の出現率を変更することで、プレイヤーにより楽しんでもらえると考えられる。

### 参考文献

[1]ちくわ,2016,[unity 入門]60 分で作るシューティングゲーム,<https://nn-hokuson.hatenablog.com/entry/2016/07/04/213231>

[2]をりはう,2023,[Unity ゲーム開発の基礎]作って学ぶブロック崩し 3-1 タイトル画面を作る,<https://note.com/worihau/n/nf7119e058287>

### 付録

- 音源
  - ▶ BGM,SE:OtoLogic,<https://otologic.jp>
- テクスチャ
  - ▶ ロケット:イラストレイン,<https://illustrain.com/>
  - ▶ エイリアン,UFO;いらすとや,<https://www.irasutoya.com/>