

## Pygame Zero を用いたゲーム制作演習 Game creation exercises using Pygame Zero

辻合 秀一†  
Hidekazu Tsujiai

### 1. はじめに

本発表は、芸術系の大学 2 年生向けにデジタルコンテンツ演習として Pygame Zero[1]を用いたゲーム制作演習の内容を報告します。受講生は、この演習までにプログラミング授業を受けていないかプログラミング授業として Processing を学んだ学生です。また、ゲーム制作は、単にプログラムを作るだけでなく、ゲームの仕様などをまとめることができるようにレジメづくりの指導も行っています。学習評価は、プレゼンテーション、デモ動画、プログラム、レジメから行っています。この演習方法は、2020 年より 3 年間継続して行っています。

### 2. デジタルコンテンツ演習

デジタルコンテンツ演習は、2006 年 4 月から始まった富山大学芸術文化学部のインタラクティブアートプログラミング教育が始まりです。インタラクティブアートプログラミング教育では、MINDSTOMS NXT と NXT-G 言語を使いガジェットを作成しました[2][3]。その後、ガジェットのサイズを小さくまとめることと汎用性を考慮し CPU 周りとして、MINDSTOMS NXT から Raspberry Pi に変更しました。それに伴い、ブロック型プログラミング言語も、NXT-G 言語から Scratch と変更しました[4]。

デジタルコンテンツ演習は、2 年生後期 1 コマ 15 回の演習です。1 コマは、90 分です。

2020 年度は、新型コロナウイルスによりリモートで演習を実施する可能性ができたため自宅の PC できる演習に変更しました。プログラミング言語も、Scratch または卒業後に役に立つと思われる Python としました。しかし、学生全員が Python を選びました。そこで、2021 年、2022 年は、プログラミング言語を Python のみとしました。

制作テーマは、ガジェットからゲームに換え、リモートでも教えやすい環境として Mu と Pygame Zero の組み合わせにしました。Python の中で Pygame Zero を選んだ理由は、ゲーム系の API の充実と資料の少なさでした[5][6]。資料が少ないことは、自分で考えること多いと想定しました。

#### 2.1 mu と Python Zero

Pygame Zero は、Raspberry Pi の OS インストール時から入っている Python です。Pygame Zero は、プログラミング環境 mu が用意されておりエディタと実行環境を提供されています (図 1)。この環境は、1 つのゲームを作り上げるに最適なもので 1 つを仕上げる演習規模には十分でした。

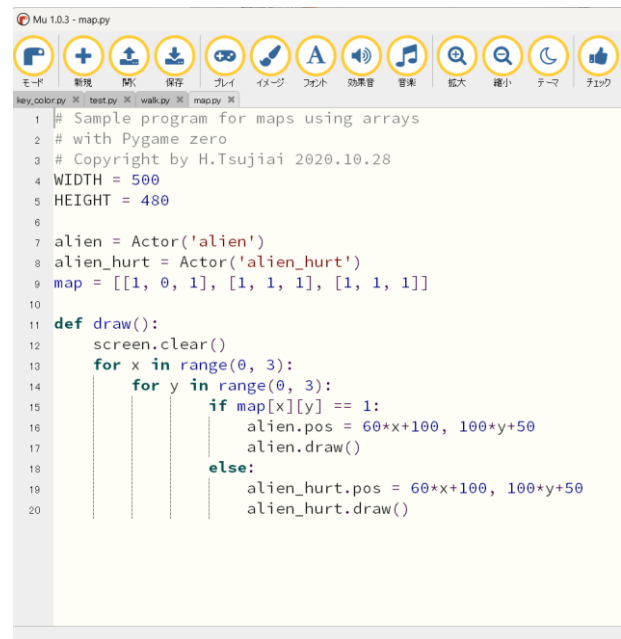
Pygame Zero は、Pygame 開発より初心者用ゲーム開発環

† 富山大学 芸術文化学系

Faculty of Art and Design, University of Toyama

境です。Pygame Zero には、Screen クラス (クリーン、Pygame の Surface と同じ)、Rect クラス (矩形)、Images クラス (PNG、GIF、JPG 画像)、Surface クラス (Pygame の Surface)、Sound クラス (WAV、OGG サウンド)、Music クラス (MP3)、Clock クラス (秒)、Actor クラス、keyboard クラス、Animation クラス、Storage クラスがあります。

また、Pygame Zero で説明のないものは Pygame のモジュールを使用します。Pygame モジュールには、BufferProxy クラス、camera クラス (カメラ)、Color クラス (RGB、CMY、HSVA、HSLA 色表現)、controller クラス (コントローラー)、cursors クラス (カーソル)、display クラス (画面制御)、draw クラス (描画)、event クラス (イベント、キュー)、font クラス (フォント)、freetype クラス (Free Type フォント)、image クラス (BMP、JPEG、PNG、TGA 画像)、key クラス (キーボード)、joystick クラス (ジョイスティック)、locals クラス (pygame 定数)、mixer クラス (サウンド)、mouse クラス (マウス)、mask クラス (画像マスク)、midi クラス (MIDI 入出力)、PixelArray クラス (サーフェス画像)、pixelcopy クラス (ピクセルコピー)、Rect クラス (矩形)、scrap クラス (クリップボード)、sndarray クラス (サウンド)、sprite クラス (スプライト)、Surface クラス (面の設定)、surfarray クラス (サーフェス配列変換)、time クラス (ミリ秒)、transform クラス (サーフェス変換)、math クラス (数学関数)、music クラス (MP3)、version クラス



```

Mu 1.0.3 - map.py
mode 新規 開く 保存 プレイ イメージ フォント 効果音 音楽 拡大 縮小 テーマ チェック
levy_color.py test.py walk.py map.py
1 # Sample program for maps using arrays
2 # with Pygame zero
3 # Copyright by H.Tsujiai 2020.10.28
4 WIDTH = 500
5 HEIGHT = 480
6
7 alien = Actor('alien')
8 alien_hurt = Actor('alien_hurt')
9 map = [[1, 0, 1], [1, 1, 1], [1, 1, 1]]
10
11 def draw():
12     screen.clear()
13     for x in range(0, 3):
14         for y in range(0, 3):
15             if map[x][y] == 1:
16                 alien.pos = 60*x+100, 100*y+50
17                 alien.draw()
18             else:
19                 alien_hurt.pos = 60*x+100, 100*y+50
20                 alien_hurt.draw()

```

図 1 Pygame Zero のプログラミング環境 mu

(バージョン) があります[7]。

Pygame Zero 上で `pygame.Color` クラスは、`pygame` を書かなくとも `Color` クラスで動作します。また、Pygame Zero 固有のものとして、`Rect` クラスの派生である `Actor` クラスや `animation` クラスがあります。

Pygame Zero の Python の文法と同じです。ただ、特有のタプルの知識は必要です。例えば、`Color` クラスは、赤を `Color.r`、緑を `Color.g`、青を `Color.b`、 $\alpha$  値を `Color.a` で記述する以外にも、`(int, int, int)` のタプルやリストで記述することもできます。

Pygame Zero で簡単な作画ならば、`Screen` クラスの `bounds()`、`clear()`、`fill()`、`blit()`、`draw.line()`、`draw.circle()`、`draw.filled_circle()`、`draw.rect()`、`draw.filled_rect()`、`draw.textbox()` 関数を使うことができます。また、音は、`Sound` クラスの `play()`、`play(loops)`、`stop()`、`get_length()` を使用し、時間は、`Clock` クラスの `schedule()`、`schedule_unique()`、`schedule_interval()`、`unscheduled()` でコントロールできます。

PygameZero では、`Rect` クラスから派生した `Actor` クラスがあり、キャラクタ設定の位置や画像を持つクラスが用意されています。例えば、図 1 の 7 行目では、`Actor` クラスとして `alien` クラスが生成され、画像として `'alien'` が初期値として与えています。また、16 行目では、`alien.pos` タプルコンストラクタに、`x` 座標と `y` 座標を与えています。`x` 座標だけ代入するならば、`alien.x` を使うことができます。図 1 に記述がありませんが、`alien` の画像は `alien.image` に保存されています。`Actor` のコンストラクタとして、`center` (画像の中心)、`opleft` (画像の左上)、`midtop` (画像の中央上)、`topright` (画像の右上)、`midleft` (画像の中央左)、`midright` (画像の中央右)、`bottomleft` (画像の左下)、`midbottom` (画像の中央下)、`bottomright` (画像の右下) があります。`pos` のデフォルトでは、`center` です。

画像ファイルやサウンドファイルは、プログラムが置かれているディレクトリ内の `images` サブディレクトリや `sounds` ディレクトリに保存します。

## 2.2 2020 年度

この年、新型コロナが流行しリモート授業に移行することも考えガジェットでも PC でもどちらでも動かすことができる教育環境を模索しゲーム制作としました。機材は、`Raspberry Pi` または `PC` としてプログラミング言語を `Scratch` または `Python` としました。

受講生は 6 名で、教師が 1 名で演習を進めました。プログラム言語は、学生と話し合い `Python` のみとなりました。

演習では、機材の取り扱いから、`Raspberry Pi` のパーツ説明からプログラミング環境、特にプログラミングは 5 回を文法などの説明を行いました。10 回目から 14 回目は、作品制作に、15 回目にプレゼンテーションを行ってもらいました。`Pygame Zero` では、画像ファイルやサウンドファイルの保存場所が決まられているので、ディレクトリの指定に迷うことはなかった。

提出物は、レジメ、`Python` プログラム等です。成績は、提出物とプレゼンテーションで行いました。レジメは、文献 8 を参考に作成させました。プレゼンテーションとれじめで、ゲームの内容や UIなどを把握しました。プログラ

ム等の提出では、プログラムは提出するが画像や音が欠損しており再現することができませんでした。

その反省から次年度より、画像ファイルやサウンドファイルの添付確認するようにしました。また、制限性ができなくても、内容が把握できるように、プレイ中の動画を添付するようにしました。

表 1 に作品一覧を示します。6 名中、プレゼンテーションまで進んだのは 5 名でした。

表 1 2020 年度デジタルコンテンツ演習

作品名	カテゴリ
ジャンピングアドベンチャー (図2)	横スクロール
戦車ゲーム	横スクロール シューティング
ホッケーゲーム	対戦
シューティングゲーム	シューティング
色合わせ	未完成



図 2 ジャンピングゲームのプレイ画面

## 2.3 2021 年度

2021 年度は、2020 年度を踏襲し、`Pygame Zero` で演習を行いました。この年は、1 名の TA が補助としてつきました。

提出物は、レジメ、`Python` プログラム (画像や音を含む)、`UML` のクラス図[9]、ゲーム中の動画としました。成績は、提出物とプレゼンテーションで行いました。

表 2 に作品一覧を示します。9 名中、プレゼンテーションまで進んだのは 7 名でした。`UML` を使って UI やコードがわかるように指導しましたが、`UML` のクラス図を描くので背一杯の状況でした。

表 2 2021 年度デジタルコンテンツ演習

作品名	カテゴリ
Space defender	シューティング
UFO を避けよう!	落ちもの
カラージュ	メディアアート
ジャンプゲーム	横スクロール
雨に濡れぬな! (図3)	アクション
シューティングゲーム	シューティング



図3 雨に濡れるな!のプレイ画面

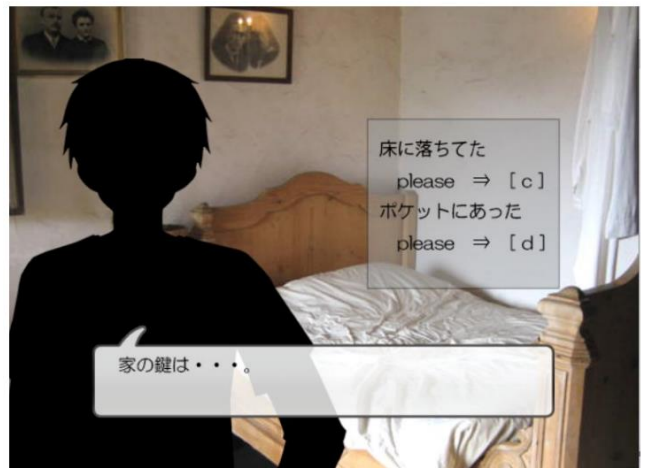


図5 コンビニでアイスが食べたいのプレイ画面

#### 2.4 2022 年度

2022 年度は、2020、2021 年度を踏襲し、Pygame Zero で演習を行いました。この年も、1 名の TA が補助としてつきました。

提出物は、昨年と同じレジメ、Python プログラム（画像や音を含む）、UML のクラス図、ゲーム中の動画としました。成績は、提出物とプレゼンテーションで行いました。

表 3 に作品一覧を示します。13 名中、プレゼンテーションまで進んだのは 6 名でした。

表 3 2022 年度デジタルコンテンツ演習

作品名	カテゴリ
月を飛び越える牛 (図 4)	横スクロール
コンビニでアイスが食べたい (図 5)	ノベル
スペースエスケープ (図 6)	シューティング
オセロ	対戦
目指せ TOP アイドル (図 7)	クリック
シューティングゲーム (図 8)	シューティング
倉庫番 (図 9)	パズル

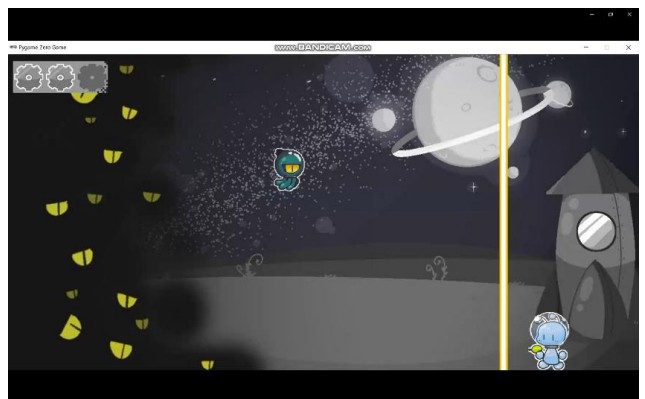


図6 スペースエスケープのプレイ画面



図4 月を飛び越える牛のプレイ画面

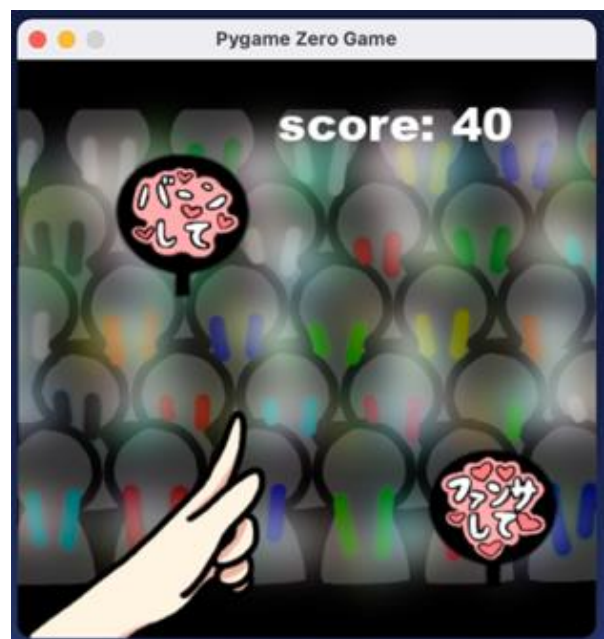


図7 目指せ TOP アイドルのプレイ画面

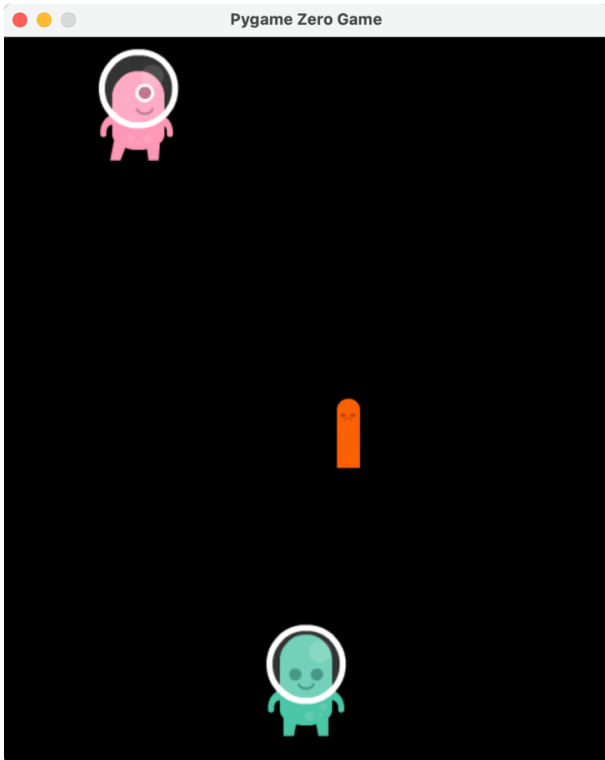


図 8 シューティングゲームのプレイ画面

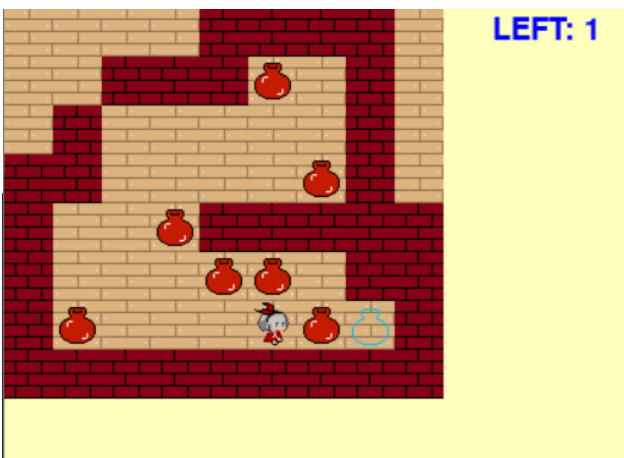


図 9 倉庫番のプレイ画面

UML のクラス図は、どのようなクラスおよび関数を使ったか、どのような使い方をしたかを把握するのに役立ちました。しかし、ゲームの UI がどのようになっているかを把握するためには UML のクラス図だけでは足りないことがわかりました。

#### 謝辞

この発表には、富山大学芸術文化学部「デジタルコンテンツ演習」受講者の協力の基でまとめることができ感謝しております。

#### 参考文献

- [1] Pygame Zero へようこそ, <https://pygame-zero.readthedocs.io/ja/latest/>(参照 2023.5.31).
- [2] 辻合 秀一, “インタラクティブアートプログラミング教育プログラム”, 第 6 回情報科学技術フォーラム一般講演論文集, 第 4 分冊, pp.391-392(2007).
- [3] 辻合 秀一, “制作を含むプログラミング教育の一考察”, 情報処理学会研究報告, Vol.2011-CE-110, No.1, pp.1-4(2011).
- [4] 辻合 秀一, “芸術系学生への Scratch を使ったコンテンツ教育”, 情報処理学会研究報告, Vol.2015-CE-131, No.15, pp.1-3(2015).
- [5] Richard Smith, “Coding Games With Pygame Zero & Python – Student workbook (2<sup>nd</sup> edition)”, Independently published (2019).
- [6] Stewart Watkiss, “Beginning Game Programming with Pygame Zero”, Apress (2020).
- [7] Pygame, <https://www.pygame.org/>(参照 2023.6.13).
- [8] 映像情報メディア学会/研究会, <https://www.ite.or.jp/content/tech-group/>(参照 2023.5.31).
- [9] 竹政 昭利ほか, “かんたん UML 入門 [改訂 2 版]”, 技術評論社 (2013).
- [10] Jaso R.Briggs, “たのしいプログラミング Python ではじめよう!”, オーム社(2014).

### 3. おわりに

芸術系大学の多様化した科目においてプログラミングの習得有無に関わらず行ったゲーム制作演習についてまとめました。1つの演習でプログラミング教育とゲームの仕様作成からプレゼンテーションまでを行うことができました。

また、プログラミングとして Python のゲームに特化した Pygame Zero を使用した演習状況をまとめることができました。Python の入門書[10]では、turtle モジュールを使って作画を試みっていますが、Pygame Zero では Actor クラスというゲームのキャラクタ設定に使い易いものが用意されており短時間でゲーム作成に辿り着くことができました。