

2D ゲームの制作

石黒滉樹 佐藤海征
宇戸隆備 高木時彦

1. 研究概要

漢字や歴史などを楽しく学べる2Dゲームを制作するために「ドット絵」「プログラム」「クイズ制作」の3つに分担して制作した。

2. 研究の具体的内容

(1) Unity について

Unity は IDE を内蔵したゲームエンジンであり 100 万人以上の開発者が利用している。開発はユニティ・テクノロジーズ iOS、Android、Android TV、Windows、Windows ストアアプリ、macOS、Linux、ウェブブラウザ、PlayStation 3・4、PlayStation Vita、Xbox360、Xbox One、Wii U そして VR/AR 向けの開発に対応している。

(2) 制作過程

制作は図 1 の流れに沿って行った。

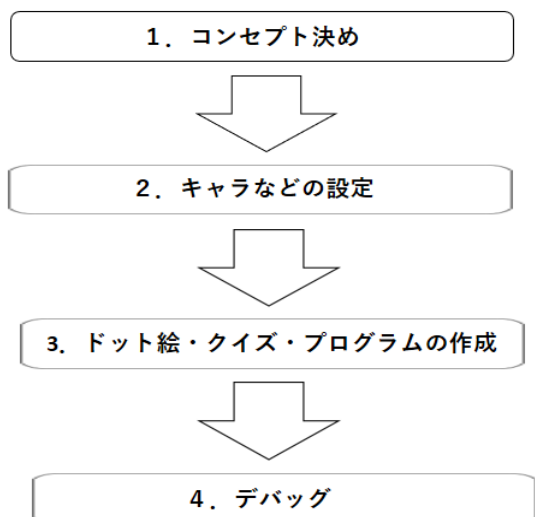


図 1 制作過程

(3) コンセプト決め

勉強をすることが好きではない人は、時間を忘れてゲームに熱中する人が多いと思った。したがって勉強の要素とゲームの要素を組み合わせたゲームにしたいと思った。

(4) キャラなどの設定

制作するゲームには、どういうキャラがいたら世界観的に合うか、また面白いのかを考え、そのキャラに合うスキルや魔法を考えた。

(5) ドット絵・クイズ・プログラムの作成

(ア) ドット絵の作成

EDGE という 256 色専用のドット絵エディタを使用して背景やキャラを作成した (図 2・3)。最初はピクセルを 32×32 で制作してゲームの仕様に合うようにリサイズした。タイトル画面は雲が流れるように見えるように複数枚の静止画を組み合わせた (図 4)。

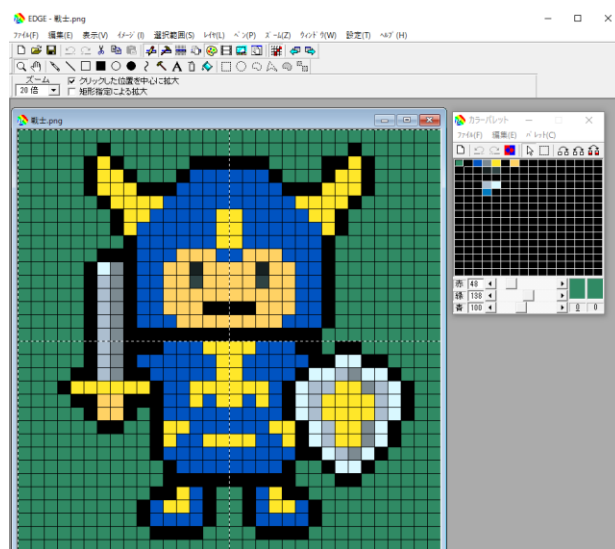


図 2 EDGE の作業画面

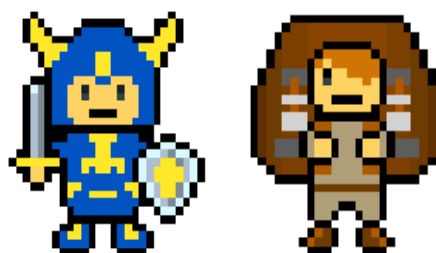


図 3 作成したドット絵



スペースキーを押してスタート

ルール説明はAキー

図4 タイトル画面

(イ) クイズの作成

クイズの作成はSPIの言語分野や歴史などの問題を参考に4択問題を作成した。各分野の問題は25問ずつ制作した。制作したクイズをメモ帳に問題と解答の二つに分けて保存しプログラムに組み込みやすいようにした(図5)。

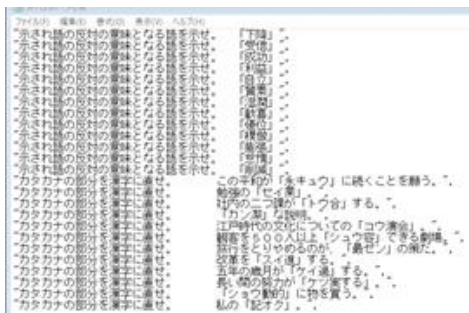


図5 クイズの原案

(ウ) プログラムの作成

ア 各シーンの作成

Unityのプロジェクトからシーンを作成。タイトル・クイズ・セレクト・バトルのシーンを作成した(図6)。

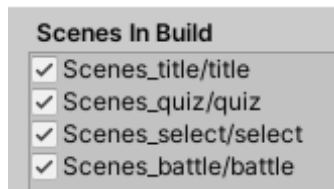


図6 シーンのビルド

イ シーン間の移動

行きたいシーンに移動するときはUnityEngineのSceneManagerにあるSceneManager.LoadScene()という関数を使い()内のシーンに移動することができる(図7)。

```

5  [using UnityEngine.SceneManagement;
6
7  @Unity スクリプト10個の参照
8  public class GoToquiz : MonoBehaviour
9  {
10     public GameObject panel;
11     public Text ptext;
12     public int reme;
13     public Button puras;
14     public Button mainas;
15     public Image s0;
16     public Sprite[] ss = new Sprite[3];
17
18     @Unity メッセージ10個の参照
19     private void Start()
20     {
21         panel.SetActive(false);
22         reme = 0;
23     }
24
25     @Unity メッセージ10個の参照
26     void Update()
27     {
28         if (Input.GetKeyDown("space"))
29         {
30             SceneManager.LoadScene("quiz");
31         }
32     }
33 }
  
```

図7 26行目がシーン移動

ウ 問題を出すプログラムの作成

問題文と解答の配列(図8)を作り、25通りの配置を作りランダム関数で配置を選択し問題がランダムに出題されるようにした(図9)。

```

28 Sentence = new string[142] {
29     "【古代文明のうち実際にない文明はどれ？】",
30     "【発祥が使用を差った国はどれ？】",
31     "【大正時代の法律の意味は？】",
32     "【戦後半期に同年に起きた？】",
33     "【徳田永年私財没は同年に発布された？】",
34     "【閉閣政治の全盛期は誰が推された？】",
35     "【明治の乱が起きたのは何年？】",
36     "【十円玉の裏に書かれている建物の名前は何？】",
37     "【幕末の乱は何年に起きた？】",
38     "【鎌倉時代は何時に起きた？】",
39     "【1467年には何が起きた？】",
40     "【フランス語で「ザ」に由来したものは何？】",
41     "【刀狩を行ったのは誰？】",
42     "【1635年に何が起きた？】",
43     "【日本の領土が拡大した？】",
44     "【戦前の時代で、戦前の甲午の戦に引かれたものを何というか？】",
45     "【古代の中国で西漢を築いた皇帝を何というか？】",
46     "【エジプト文明の特徴として正しいのはどれか？】",
47     "【朝鮮半島に起こり、朝鮮を治めた国はどれか？】",
48     "【アビラ、イノの戦は誰が勝った？】",
49     "【大航海時代最盛期だったものはどれか？】",
50     "【この中で、仏教を国教として採られた国はどれか？】",
51     "【1922年、イタリアの政権を握ったファシスト党の党首は誰？】",
52     "【第一次世界大戦後、国際連盟の設立を提唱したアメリカの大統領は誰か？】",
53     "【アメリカ合衆国の初代大統領は誰？】",
54     "【1911年中国で起こった辛亥革命を指導者とした人を何という？】",
55     "【1881年にアフリカの露骨を奪ってインドへの覇権を争った人物は？】",
56     "【19世紀後半に起きた、産業革命の革命はどれか？】",
57     "【アメリカで自由貿易と保護貿易をめぐる対立によって起こった戦争はどれか？】",
58     "【インドで独立運動を行ったのは誰？】",
59     "【ニューヨーク株式市場の株価大暴落による経済の混乱を何というか？】",
60     "【アインシュタインが相対性理論を発表した国はどれか？】",
61     "【現在のEはローマ numeralの何を意味しているか？】",
62     "【「ヨロロツの火災」呼ばれたのはどれ？】",
63     "【資本主義が生み出した社会問題は何？】",
64     "【フランス語で「ザ」に由来した職業はどれか？】",
65     "【幕末の乱は誰が勝った？】",
66     "【孔子が生まれた国はどれか？】",
67     "【アヘン戦争を引き起こしたアヘンはこの国のものか？】",
68     "【オランダの東印度会社はどの国の領土か？】",
  
```

図8 クイズを配列の形にした

```

479 int arandum = Random.Range(0, 24);
480 switch (arandum)
481 {
482
483     case 0:
484         a = 0;
485         s = 1;
486         d = 2;
487         f = 3;
488         break;
489     case 1:
490         a = 0;
491         s = 1;
492         d = 3;
493         f = 2;
494         break;
495     case 2:
  
```

図9 解答の位置のランダム化

エ キャラを選択するプログラムの作成
それぞれのキャラごとにステータスの配列を制作してボタンを押すと自分のステ

ータスに代入してキャラを選択できるようにした (図 10)。

```

53  playername = new string[6] { "戦士", "盗賊", "ゴリラ", "魔法使い", "聖騎士", "商人" };
54  //敵の選択
55  float tap;
56  System.Random r = new System.Random(quiz.SP);
57  tap = r.Next(6);
58  switch (tap)
59  {
60
61  case 0:
62      float[] STA0 = { 100, 100, 10, 10, 10, 10, 10, 0 };
63      ENEMYSTATUS = STA0;
64      break;
65
66  case 1:
67      float[] STA1 = { 90, 110, 10, 10, 5, 5, 20, 1 };
68      ENEMYSTATUS = STA1;
69      break;
70
71  case 2:
72      float[] STA2 = { 200, 0, 30, 0, 120, -100, 10, 0, 2 };
73      ENEMYSTATUS = STA2;
74      break;
75
76  case 3:
77      float[] STA3 = { 80, 120, 0, 20, 5, 20, 5, 10, 3 };
78      ENEMYSTATUS = STA3;
79      break;
80
81  case 4:
82      float[] STA4 = { 100, 100, 10, 10, 15, 15, 5, 5, 4 };
83      ENEMYSTATUS = STA4;
84      break;
85
86  case 5:
87      float[] STA5 = { 110, 90, 5, 5, 5, 5, 35, 5 };
88      ENEMYSTATUS = STA5;
89      break;
90
91  default:
92      break;

```

図 10 SWITCH 文で各キャラを選択

また、ステータスも押したボタンに対応したステータスが増減するようプログラムも作成した (図 11)。

```

136  if (SPP)
137  {
138      if (STT <= 7)
139      {
140          time -= 0.1f;
141          if (time < 0)
142          {
143              if (sp > 0 && STATUS[STT] < maxValue)
144              {
145                  STATUS[STT]++;
146                  sp--;
147              }
148          }
149      }
150  }
151
152  if (SPM)
153  {
154      if (STT <= 7)
155      {
156          time -= 0.1f;
157          if (time < 0)
158          {
159              if (STATUS[STT] > minValue)
160              {
161                  STATUS[STT]--;
162                  sp++;
163              }
164          }
165      }
166  }
167
168  }
169
170  }
171
172  }

```

図 11 146, 164 で数値の増減

(オ) ターン制バトルのプログラムの作成

自分のターンと相手のターンを区別できるようにブール関数を使用して判断できるようにした。自分のターンと相手のターンの区切りの時にディレイをかけて、ターンが変わるのがわかりやすくした (図 13)。自分のターンにはボタンを使って攻撃できるようにし、相手のターンの時はランダムに物理攻撃・スキル攻撃・魔法攻撃をしてくるように設定した (図 12)。また、キャラごとに異なるスキル攻撃を設定しバトルに多様性を持たせたい

だけでなくゲームとして面白くなるようにした。

```

325  switch (ii)
326  {
327      //魔法攻撃
328      case 0:
329      case 1:
330      case 2:
331      case 3:
332      case 4:
333          kougeki_e_mahou();
334          break;
335      //物理攻撃
336      case 5:
337      case 6:
338      case 7:
339      case 8:
340      case 9:
341      case 10:
342      case 11:
343      case 12:
344      case 13:
345      case 14:
346      case 15:
347      case 16:
348      case 17:
349      case 18:
350          kougeki_e_buturi();
351          break;
352      //スキル
353      case 19:

```

図 12 敵の攻撃のランダム化

```

257  count += 0.01f;
258
259  if (count > 1f)
260  {
261      if (eee == 0)
262      {
263          Enemytrun();
264          eee += 1;
265      }
266
267  }
268
269  if (count > 2f)
270  {
271      //HP、MPの回復
272      if (MPS_P.value < MPS_P.maxValue)
273      {
274          if (STATUS_P[B] == 3)
275              MPS_P.value += MPS_P.maxValue * 0.2f;
276          else
277              MPS_P.value += MPS_P.maxValue * 0.1f;
278      }
279      if (MPS_E.value < MPS_E.maxValue)
280      {
281          if (STATUS_E[B] == 3)
282              MPS_E.value += MPS_E.maxValue * 0.2f;
283          else
284              MPS_E.value += MPS_E.maxValue * 0.1f;
285      }
286      count = 0;
287      eee = 0;
288      mytrun = true;
289  }
290
291  }
292
293  }
294

```

図 13 257~267 でディレイをかけている

(カ) サウンド・エフェクトの実装

各キャラにあったエフェクト、サウンド実装することでゲームの演出が単調にならないようにした (図 14)。

```

395 //効果音&エフェクト
396 switch (STATUS_P[8])
397 {
398     case 0:
399         p.SetTrigger("ssbk");
400         audios.PlayOneShot(Ssbk);
401         break;
402     case 1:
403         p.SetTrigger("tzbk");
404         audios.PlayOneShot(Tzbk);
405         break;
406     case 2:
407         p.SetTrigger("grbk");
408         audios.PlayOneShot(Grbk);
409         break;
410     case 3:
411         p.SetTrigger("mtbk");
412         audios.PlayOneShot(Mtbk);
413         break;
414     case 4:
415         p.SetTrigger("skbk");
416         audios.PlayOneShot(Skbk);
417         break;
418     case 5:
419         p.SetTrigger("snbk");
420         audios.PlayOneShot(Snbk);
421         break;

```

図 1 4 SWITCH 文でキャラごとに音を実装 (キ)デバッグ

ダメージ計算の符号のミスやHPバーMPバーを自由に操作できてしまうバグ、エフェクトとサウンドが一致していないバグ、相手のスキル名が別のスキル名になってしまうバグなどを修正した。

3. 研究のまとめ

佐藤

今回のゲーム作成において、最初は cocos2DX で作成しようとした。しかし、動作せず Unity で作成しようということになり、最初の開発環境選びからつまづいた感じがした。ゲーム全体の構成を考え、チーム制作をしようとする Unity で共同開発ができず、結局、個人製作が多くなってしまった。制作していくと様々な設定が多く、四苦八苦しながらひとまずゲームの流れにそって動作するようになった。動作するようになってからもサウンドなどの設定が大変で時間がかかった。さらに、オンライン対戦ができるようにしようと思っていたがセキュリティ上の理由から断念した。何千行とあるプログラムを作ったこともなく、シーン間での変数の振る舞いや、データ型の違いで何度もエラーを出したりしながら制作していった。構想通りの動作をさせるためにどのようなプログラムを作るかわからず調べたが、その通りにしても設定な

どの影響で動作しないこともあり、結局は参考程度にしながら理解していくことになった。Unity で 2D ゲーム制作を通じて、Unity、C# に対する理解が深まったと思う。ほかにも、チーム間の連携が取れていない時もあったのでまた何かでチーム制作をするときはしっかりと報・連・相をしていきたいと思った。

石黒

今回のゲーム作りではコロナウイルスによる影響で各メンバーが同時に製作できる環境やタイミングがあまりとれず、製作当初予定していたものより進まなかったりして、大変なことだらけだったが最低限形になり、ゲームが完成して良かったと思った。

宇戸

僕は絵の担当をした。EDGE というソフトを使用してドット絵の制作を行った。ドット絵の制作において一番大変だったのは、絵の心得がないので何から手を付けていいかわからず、その点がとても苦戦した。その中で勉強し、模索しながらドット絵を制作した。課題研究を通してドット絵やゲームの製作方法を学び、それらを将来に活かしていきたいと思った。

高木

僕はゲームの素材を集めてきたりクイズを SPI の問題などを基に制作した。ゲームの素材はサウンド、エフェクトを集めた。サウンドはオープニングやキャラ選択などの各場面の BGM をその場面に合ったものを探し選ぶのは大変だった。クイズの制作は SPI の問題集などから各教科の問題を探したりして制作した。

4. 参考文献

https://www.youtube.com/channel/UCDunz_CPkqkQT51jKXcYkhg