

C言語初めてのゲーム(走り抜けジャンプ)-part3

対象

前回のC言語初めてのゲーム(走る抜けジャンプ)-part2を終えている人が対象となっています。終えてない人は以下のリンクから飛んでやってください。
[C言語初めてのゲーム\(走り抜けジャンプ\)-part2](#)

構成

- [No.5 ゲーム本編](#)
- [No.5-1 必要な関数と準備](#)
- [No.5-2 描画、読み込み](#)

本編

今回は前回の続きからゲームの本編を作り、ゲームの完成を目指します。

No.5 ゲーム本編

今回はゲームの中身を作ります。ゲームはタイトル通りでGoogle Chromeのエラー画面で遊べるあのゲームみたいのを作ります。ただ作るとはいきこんでますが作り方なんて何通りでもありますし私の作り方にひどい部分があるかもしれませんご了承ください。私が伝えたいことはゲームの基本とゲームは簡単に作れるということです。

No.5-1 必要な関数と準備

今までと同様、まず最低でも必要となる関数を用意します。最低限と称したようにこの後で関数は追加していくと思われます。

```
void Initialize_Game(void);           // ゲームの要素の初期化
void Game_Update(void);               // ゲームの描画と操作の実体を管理
void Game_Draw(void);                // ゲームの描画の中身を管理
void Game_Process(void);              // ゲームの操作の中身を管理
void Delete_Game(void);              // ゲームの要素の削除
```

特にそれぞれの説明はいらないと思います。ではソースコードに組み込んでみましょう。

```
1. #include "DxLib.h"
```

```
2.  
3. // 初期化  
4. void Initialize_Title(void); // タイトルの要素の初期化  
5. void Initialize_Game(void); // ゲームの要素の初期化  
6. void Initialize_End(void); // エンディングの要素の初期化  
7.  
8. // 削除  
9. void Delete_Title(void); // タイトルの要素の削除  
10. void Delete_Game(void); // ゲームの要素の削除  
11. void Delete_End(void); // エンディングの要素の削除  
12.  
13. // タイトルの実体  
14. void Title_Update(void); // タイトル描画と操作の実体を管理  
15. void Title_Draw(void); // タイトルの描画の中身を管理  
16. void Title_Process(void); // タイトルの操作の中身を管理  
17.  
18. // ゲームの実体  
19. void Game_Update(void); // ゲームの描画と操作の実体を管理  
20. void Game_Draw(void); // ゲームの描画の中身を管理  
21. void Game_Process(void); // ゲームの操作の中身を管理  
22.  
23. // エンディングの実体  
24. void End_Update(void); // エンディングの描画と操作の実体を管理  
25. void End_Draw(void); // エンディングの描画の中身を管理  
26. void End_Process(void); // エンディングの操作の中身を管理  
27.  
28. // タイトル要素  
29. // 描画  
30. int gr_title_back; // 背景  
31. int gr_title_cursor; // カーソル絵  
32. int gr_title_start; // 始めるアイコン画像  
33. int gr_title_end; // 終わるアイコン画像  
34. // 操作  
35. int title_cursor_y; // カーソル動作  
36.  
37. // エンディング要素  
38. // 描画  
39. int gr_end_back; // 背景  
40.  
41. // シーン  
42. int scene;  
43.  
44. // 終了  
45. bool endflag;  
46.  
47. int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
```

```
48. {
49.     ChangeWindowMode(TRUE);           // ウィンドウモード変更
50.     DxLib_Init();                  // 初期化
51.     SetDrawScreen(DX_SCREEN_BACK); // 裏画面設定
52.
53.     scene = 0;
54.     endflag = false;
55.
56.     Initialize_Title();           // タイトルの初期化の呼び出し
57.     Initialize_Game();           // ゲームの初期化の呼び出し
58.     Initialize_End();           // エンディングの初期化の呼び出し
59.
60.     while( ScreenFlip() == 0 && ProcessMessage() == 0 && ClearDrawScreen() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0 && endflag == false)
61.     {
62.         if(scene == 0)
63.         {
64.             Title_Update();           // タイトル
65.         }
66.         else if(scene == 1)
67.         {
68.             Game_Update();           // ゲーム
69.         }
70.         else if(scene == 2)
71.         {
72.             End_Update();           // エンディング
73.         }
74.     }
75.
76.     Delete_Title();                // タイトルの削除
77.     Delete_Game();                // ゲームの削除
78.     Delete_End();                // エンディングの削除
79.
80.     DxLib_End(); // DXライブラリ終了処理
81.     return 0;
82. }
83.
84. // タイトルの初期化
85. void Initialize_Title()
86. {
87.     // 描画
88.     gr_title_back = LoadGraph("media\\title_background.png");
89.     gr_title_cursor = LoadGraph("media\\title_cursor.png");
90.     gr_title_start = LoadGraph("media\\title_start.png");
91.     gr_title_end = LoadGraph("media\\title_end.png");
92.
93.     title_cursor_y = 0;
```

```
94. }
95.
96. // タイトルの削除
97. void Delete_Title()
98. {
99.     DeleteGraph(gr_title_back);
100.    DeleteGraph(gr_title_cursor);
101.    DeleteGraph(gr_title_start);
102.    DeleteGraph(gr_title_end);
103. }
104.
105. // タイトルのすべて
106. void Title_Update()
107. {
108.     Title_Draw();
109.     Title_Process();
110. }
111.
112. // タイトルの描画
113. void Title_Draw()
114. {
115.     DrawGraph(0, 0, gr_title_back, false);
116.     DrawGraph(240, 300 + (50 * title_cursor_y), gr_title_cursor, true);
117.     DrawGraph(290, 300, gr_title_start, true);
118.     DrawGraph(290, 350, gr_title_end, true);
119. }
120.
121. // タイトルの操作
122. void Title_Process()
123. {
124.     if(CheckHitKey(KEY_INPUT_UP) == 1)
125.     {
126.         if(title_cursor_y != 0)
127.         {
128.             title_cursor_y++;
129.         }
130.     }
131.     if(CheckHitKey(KEY_INPUT_DOWN) == 1)
132.     {
133.         if(title_cursor_y != 1)
134.         {
135.             title_cursor_y--;
136.         }
137.     }
138.
139.     if(CheckHitKey(KEY_INPUT_Z) == 1)
```

```
140.    {
141.        if(title_cursor_y == 0)
142.        {
143.            scene = 2;      // 本来はゲームに移るため1だが処理がないので2とする
144.        }
145.        else if(title_cursor_y == 1)
146.        {
147.            endflag = true;
148.        }
149.    }
150. }
151.
152. // ゲームの初期化
153. void Initialize_Game()
154. {
155.
156. }
157.
158. // ゲームの削除
159. void Delete_Game()
160. {
161.
162. }
163.
164. // ゲームのすべて
165. void Game_Update()
166. {
167.
168. }
169.
170. // ゲームの描画
171. void Game_Draw()
172. {
173.
174. }
175.
176. // ゲームの操作
177. void Game_Process()
178. {
179.
180. }
181.
182. // エンディングの初期化
183. void Initialize_End()
184. {
185.     // 描画
```

```
186.     gr_end_back = LoadGraph("media/end_background.png");
187. }
188.
189. // エンディングの削除
190. void Delete_End()
191. {
192.     DeleteGraph(gr_end_back);
193. }
194.
195. // エンディングのすべて
196. void End_Update()
197. {
198.     End_Draw();
199.     End_Process();
200. }
201.
202. // エンディングの描画
203. void End_Draw()
204. {
205.     DrawGraph(0, 0, gr_end_back, false);
206. }
207.
208. // エンディングの操作
209. void End_Process()
210. {
211.     if (CheckHitKey(KEY_INPUT_Z) == 1)
212.     {
213.         scene = 0;
214.     }
215. }
```

ゲーム部分以外も少しいじりました。要はシーンの順番を変えただけです。
次はゲームの画像の読み込みを作ります。ちょっとずつ進みましょう。ちょっとずつ。

No.5-2 描画、読み込み

今回のゲームに必要な画像は以下のものが予想されます。

キャラクター
障害物
背景

以上です。それぞれ一枚だけいいです。少ないように思えますがこれだけで十分です。

では、それらを扱う変数を用意します。

```
int gr_game_back;           // 背景
int gr_game_character;      // キャラクター
int gr_game_enemy;          // 敵
```

特に特別なことはないので今までのようこの変数に画像を読み込ませ、表示させます。

```
1. #include "DxLib.h"
2.
3. // 初期化
4. void Initialize_Title(void);    // タイトルの要素の初期化
5. void Initialize_Game(void);     // ゲームの要素の初期化
6. void Initialize_End(void);      // エンディングの要素の初期化
7.
8. // 削除
9. void Delete_Title(void);        // タイトルの要素の削除
10. void Delete_Game(void);        // ゲームの要素の削除
11. void Delete_End(void);         // エンディングの要素の削除
12.
13. // タイトルの実体
14. void Title_Update(void);        // タイトル描画と操作の実体を管理
15. void Title_Draw(void);         // タイトルの描画の中身を管理
16. void Title_Process(void);       // タイトルの操作の中身を管理
17.
18. // ゲームの実体
19. void Game_Update(void);         // ゲームの描画と操作の実体を管理
20. void Game_Draw(void);          // ゲームの描画の中身を管理
21. void Game_Process(void);        // ゲームの操作の中身を管理
22.
23. // エンディングの実体
24. void End_Update(void);         // エンディングの描画と操作の実体を管理
25. void End_Draw(void);           // エンディングの描画の中身を管理
26. void End_Process(void);        // エンディングの操作の中身を管理
27.
28. // タイトル要素
29. // 描画
30. int gr_title_back;            // 背景
31. int gr_title_cursor;          // カーソル絵
32. int gr_title_start;           // 始めるアイコン画像
33. int gr_title_end;             // 終わるアイコン画像
34. // 操作
35. int title_cursor_y;           // カーソル動作
36.
37. // ゲーム要素
```

```
38. // 描画
39. int gr_game_back;           // 背景
40. int gr_game_character;     // キャラクター
41. int gr_game_enemy;         // 敵
42.
43. // エンディング要素
44. // 描画
45. int gr_end_back;           // 背景
46.
47. // シーン
48. int scene;
49.
50. // 終了
51. bool endflag;
52.
53. int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
54. {
55.     ChangeWindowMode(TRUE);           // ウィンドウモード変更
56.     DxLib_Init();                  // 初期化
57.     SetDrawScreen(DX_SCREEN_BACK);   // 裏画面設定
58.
59.     scene = 0;
60.     endflag = false;
61.
62.     Initialize_Title();           // タイトルの初期化の呼び出し
63.     Initialize_Game();           // ゲームの初期化の呼び出し
64.     Initialize_End();           // エンディングの初期化の呼び出し
65.
66.     while( ScreenFlip() == 0 && ProcessMessage() == 0 && ClearDrawScreen() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0 && endflag == false)
67.     {
68.         if(scene == 0)
69.         {
70.             Title_Update();           // タイトル
71.         }
72.         else if(scene == 1)
73.         {
74.             Game_Update();           // ゲーム
75.         }
76.         else if(scene == 2)
77.         {
78.             End_Update();           // エンディング
79.         }
80.     }
81.
82.     Delete_Title();               // タイトルの削除
83.     Delete_Game();               // ゲームの削除
```

```
84.     Delete_End();           // エンディングの削除
85.
86.     DxLib_End(); // DXライブラリ終了処理
87.     return 0;
88. }
89.
90. // タイトルの初期化
91. void Initialize_Title()
92. {
93.     // 描画
94.     gr_title_back = LoadGraph("media\\title_background.png");
95.     gr_title_cursor = LoadGraph("media\\title_cursor.png");
96.     gr_title_start = LoadGraph("media\\title_start.png");
97.     gr_title_end = LoadGraph("media\\title_end.png");
98.
99.     title_cursor_y = 0;
100. }
101.
102. // タイトルの削除
103. void Delete_Title()
104. {
105.     DeleteGraph(gr_title_back);
106.     DeleteGraph(gr_title_cursor);
107.     DeleteGraph(gr_title_start);
108.     DeleteGraph(gr_title_end);
109. }
110.
111. // タイトルのすべて
112. void Title_Update()
113. {
114.     Title_Draw();
115.     Title_Process();
116. }
117.
118. // タイトルの描画
119. void Title_Draw()
120. {
121.     DrawGraph(0, 0, gr_title_back, false);
122.     DrawGraph(240, 300 + (50 * title_cursor_y), gr_title_cursor, true);
123.     DrawGraph(290, 300, gr_title_start, true);
124.     DrawGraph(290, 350, gr_title_end, true);
125. }
126.
127. // タイトルの操作
128. void Title_Process()
129. {
```

```
130.     if(CheckHitKey(KEY_INPUT_UP) == 1)
131.     {
132.         if(title_cursor_y != 0)
133.         {
134.             title_cursor_y++;
135.         }
136.     }
137.     if(CheckHitKey(KEY_INPUT_DOWN) == 1)
138.     {
139.         if(title_cursor_y != 1)
140.         {
141.             title_cursor_y--;
142.         }
143.     }
144.
145.     if(CheckHitKey(KEY_INPUT_Z) == 1)
146.     {
147.         if(title_cursor_y == 0)
148.         {
149.             scene = 1;
150.         }
151.         else if(title_cursor_y == 1)
152.         {
153.             endflag = true;
154.         }
155.     }
156. }
157.
158. // ゲームの初期化
159. void Initialize_Game()
160. {
161.     // 描画
162.     gr_game_back = LoadGraph("media\game_back.png");
163.     gr_game_character = LoadGraph("media\character.png");
164.     gr_game_enemy = LoadGraph("media\enemy.png");
165. }
166.
167. // ゲームの削除
168. void Delete_Game()
169. {
170.     DeleteGraph(gr_game_back);
171.     DeleteGraph(gr_game_character);
172.     DeleteGraph(gr_game_enemy);
173. }
174.
175. // ゲームのすべて
```

```
176. void Game_Update()
177. {
178.     Game_Draw();
179.     Game_Process();
180. }
181.
182. // ゲームの描画
183. void Game_Draw()
184. {
185.     DrawGraph(0, 0, gr_game_back, false);
186.     DrawGraph(50, 240, gr_game_character, true);
187.     DrawGraph(400, 240, gr_game_enemy, true);
188. }
189.
190. // ゲームの操作
191. void Game_Process()
192. {
193.     if(CheckHitKey(KEY_INPUT_Z) == 1)
194.     {
195.         scene = 2;
196.     }
197. }
198.
199. // エンディングの初期化
200. void Initialize_End()
201. {
202.     // 描画
203.     gr_end_back = LoadGraph("media\end_background.png");
204. }
205.
206. // エンディングの削除
207. void Delete_End()
208. {
209.     DeleteGraph(gr_end_back);
210. }
211.
212. // エンディングのすべて
213. void End_Update()
214. {
215.     End_Draw();
216.     End_Process();
217. }
218.
219. // エンディングの描画
220. void End_Draw()
221. {
```

```
222.     DrawGraph(0, 0, gr_end_back, false);  
223. }  
224.  
225. // エンディングの操作  
226. void End_Process()  
227. {  
228.     if (CheckHitKey(KEY_INPUT_Z) == 1)  
229.     {  
230.         scene = 0;  
231.     }  
232. }
```

この処理で必要な画像を出力しました。ただ出力しただけなので何もできない状態です。シーンはループ移動できるようにしました。

あと次回からソースコードの変更箇所は変数宣言とゲーム関数だけなので必要なものは以下のソースコードでは省略します。自分で必要と思われる場所に追加してください。

[C言語初めてのゲーム\(走り抜けジャンプ\)-part4](#)

Copyright (C) 2002-2009 Xiph.org Foundation Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
libpng Copyright (C) 2004, 2006-2012 Glenn Randers-Pehrson.
zlib Copyright (C) 1995-2012 Jean-loup Gailly and Mark Adler.
DX Library Copyright (C) 2001-2014 Takumi Yamada.