

```
1 #include "Game.h"
2 #include "SceneMgr.h"
3 #include "DxLib.h"
4
5
6 /**
7  グローバル変数
8  */
9 // 修正箇所
10 struct Object Nums[NUM_MAX]; // ナンバー
11 int NumsImage[NUM_MAX]; // ナンバー画像 (格納用配列)
12
13 int EndTime; // 終了タイム
14 static int mImageHandle; //画像ハンドル格納用変数
15
16 //初期化
17 void Game_Initialize() {
18     mImageHandle = LoadGraph(TEXT("images/Scene_Game.png")); //画像のロード
19     LoadDivGraph("images/num.png", 25, 5, 5, 80, 80, NumsImage); //ナンバー画像
20
21     for (int i = 0; i < NUM_MAX; i++) {
22         Nums[i].flg = 1;
23         Nums[i].x = 170 + (i % 5) * 82;
24         Nums[i].y = 40 + i / 5 * 82;
25         Nums[i].w = 80;
26         Nums[i].h = 80;
27         Nums[i].image = i;
28     }
29 }
30
31 //終了処理
32 void Game_Finalize() {
33     DeleteGraph(mImageHandle); //画像の解放
34 }
35
36 //更新
37 void Game_Update() {
38     if (CheckHitKey(KEY_INPUT_ESCAPE) != 0) { //Escキーが押されていたら
39
40         SceneMgr_ChangeScene(eScene_Clear); //シーンをゲーム画面に変更
41     }
42 }
43
44 //描画
45 void Game_Draw() {
46     DrawGraph(0, 0, mImageHandle, FALSE);
47     // 修正箇所
48     //DrawString(20, 380, TEXT("ゲーム画面です。"), GetColor(136, 136, 255));
49     //DrawString(20, 400, TEXT("Escキーを押すとメニュー画面に戻ります。"), GetColor(136, 136, 255));
50     for (int i = 0; i < NUM_MAX; i++) {
51         if (Nums[i].flg == 1) {
52             DrawGraph(Nums[i].x, Nums[i].y, NumsImage[Nums[i].image], TRUE); //ナンバーを
53             描画
54         }
55     }
56 }
```