

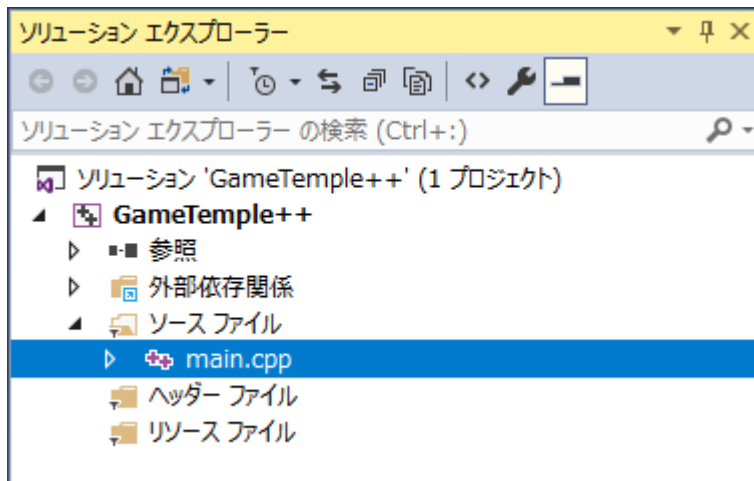
## 3日で分かる C++言語でのゲーム開発 実践①

## ■はじめに

C++言語でプログラミングするには実践が一番です。まずはゲームのひな型を作っていきます。

## ■工程1

新規プロジェクトを作成して、開発ができる様に設定を行う(DxLib など)。



main.cpp

```

1  #include "DxLib.h"
2
3  #define SCREEN_SIZE_X 800
4  #define SCREEN_SIZE_Y 600
5
6  int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
7  {
8      SetWindowText("C++Sample");
9      SetGraphMode(SCREEN_SIZE_X, SCREEN_SIZE_Y, 16); // 画面サイズ、色設定
10     ChangeWindowMode(true); // true:window/false:フルスクリーン
11     if (DxLib_Init() == -1) {
12         return -1; // DXライブラリ初期化処理
13     }
14     SetDrawScreen(DX_SCREEN_BACK);
15
16     // ----- ゲームループ -----
17     while (ProcessMessage() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0)
18     {
19         ClsDrawScreen();
20
21         ScreenFlip();
22     }
23     DxLib_End();
24     return 0;
25 }
26

```

## ■ 工程2

画面遷移を行うゲームループをクラスを使って実装する。

「GameTask.h」「GameTask.cpp」を作成し、GameTask クラスを実装していく。



GameTask.h

```
1  #pragma once
2
3  // 画面遷移用
4  enum GAME_MODE {
5      GAME_INIT,
6      GAME_TITLE,
7      GAME_MAIN,
8      GAME_OVER,
9      GAME_CLEAR
10 };
11
12 class GameTask
13 {
14 public:
15     GAME_MODE gameMode; // 状態管理を行う変数
16
17     GameTask(); // デフォルトコンストラクタ
18     ~GameTask(); // デストラクタ
19
20     int SystemInit(void);
21 };
22
```

GameTask.cpp

```
1  #include "DxLib.h"
2  #include "GameTask.h"
3
4  GameTask::GameTask()
5  {
6      SystemInit();
7  }
8
9  GameTask::~GameTask()
10 {
11 }
12
13 int GameTask::SystemInit()
14 {
15     return 0;
16 }
```

## ■ 工程3

システム初期化の部分を GameTask::SystemInit()に移して、ゲームループオブジェクトで運用する。

main.cpp

```
1  #include "DxLib.h"
2
3  #define SCREEN_SIZE_X 800
4  #define SCREEN_SIZE_Y 600
5
6  int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
7  {
8      SetWindowText("C++Sample");
9      SetGraphMode(SCREEN_SIZE_X, SCREEN_SIZE_Y, 16); // 画面サイズ、色設定
10     ChangeWindowMode(true); // true:window/false:フルスクリーン
11     if (DxLib_Init() == -1) {
12         return -1; // DXライブラリ初期化处理
13     }
14     SetDrawScreen(DX_SCREEN_BACK);
15
16     // ----- ゲームループ -----
17     while (ProcessMessage() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0)
18     {
19         ClsDrawScreen();
20
21         ScreenFlip();
22     }
23     DxLib_End();
24     return 0;
25 }
26
```

※画面サイズのマクロも GameTask に移動

※移動

GameTask.cpp

```
#include "DxLib.h"
#include "GameTask.h"

#define SCREEN_SIZE_X 800
#define SCREEN_SIZE_Y 600

int GameTask::SystemInit()
{
    SetWindowText("C++Sample");
    SetGraphMode(SCREEN_SIZE_X, SCREEN_SIZE_Y, 16); // 画面サイズ、色設定
    ChangeWindowMode(true); // true:window/false:フルスクリーン
    if (DxLib_Init() == -1) {
        return -1; // DXライブラリ初期化处理
    }
    SetDrawScreen(DX_SCREEN_BACK);

    return 0;
}
```

この状態でコンパイルをするとエラーがでます。

```
1 #include "DxLib.h"
2
3 int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
4 {
5     // ----- ゲームループ
6     while (ProcessMessage() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0)
7     {
8         ClsDrawScreen();
9
10        ScreenFlip();
11    }
12    DxLib_End();
13    return 0;
14 }
15
```

例外がスローされました  
0x00B5EF53 で例外がスローされました (GameTemple++.exe 内):  
0xC0000005: 場所 0x00000014 の読み取り中にアクセス違反が発生しました

理由は分かりますか？

これは、まだ DxLib の初期化が実行されていない状況で DxLib の命令を使用しているからです。  
では、実際に GameTask オブジェクトを生成してシステム初期化が実行される様に見てみます。

## ■ 工程4

GameTask オブジェクト実装する。

```
1 #include "DxLib.h"
2 #include "GameTask.h"
3
4 int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
5 {
6     GameTask* gameTask;
7     gameTask = new GameTask();
8
9     // ----- ゲームループ
10    while (ProcessMessage() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0)
11    {
12        ClsDrawScreen();
13
14        gameTask->
15
16        ScreenFlip();
17    }
18    DxLib_End();
19    return 0;
20 }
```

GameTask 型のポインター「gameTask」を作成し GameTask オブジェクトを new して作成します。  
オブジェクトの生成の事をインスタンスと言います。  
インスタンスのメンバ関数(クラスメソッド)へのアクセスは、gameTask->で行います。

## ■ 工程5

GameTask クラスに、各画面遷移の関数を追加しましょう。

GameTask.h

```
int GameInit(void);
int GameTitle(void);
int GameMain(void);
int GameOver(void);
int GameClear(void);
```

GameTask.cpp

```
int GameTask::GameOver()
{
    DrawString(0, 0, "GAME_OVER", 0xffff00);
    return 0;
}

int GameTask::GameClear()
{
    DrawString(0, 0, "GAME_CLEAR", 0xffff00);
    return 0;
}
```

※GameTitle()、GameMain()なども同様

## ■ 工程6

GameTask クラスに、各画面遷移の関数を集中管理する Update 関数を作成し、その Update 関数の中で switch 文を使って画面遷移を達成する様にしていきたいと思います。

```
1  #include "DxLib.h"
2  #include "GameTask.h"
3
4  #define SCREEN_SIZE_X 800
5  #define SCREEN_SIZE_Y 600
6
7  int GameTask::Update()
8  {
9      switch (gameMode) {
10         case GAME_INIT:
11             gameMode = GAME_TITLE;
12             break;
13         case GAME_TITLE:
14             break;
15         }
16         return 0;
17     }
```

## ■課題1

GameTask クラスに、各画面遷移の関数を集中管理する Update 関数を作成し、その Update 関数の中で switch 文を使って (if 文や関数ポインタでも良いです) 画面遷移を達成する様にいきましょう。