

```

#include "GameMainScene.h"

#include "DxLib.h"

#include "Stage.h"

#include "SceneManager.h"


/*****

* マクロ定義

*****/

#define TIMELIMIT                (3600 * 3)    // 制限時間

#define NUMBER_IMAGE_MAX (10)        // 数字画像数


/*****

* 型定義

*****/


/*****

* 変数宣言

*****/

int GameScore;

int GameLevel;

int GameMission;

int GameTime;

int GameCount;    // 初期化されないようにするためのカウンント

int ReStartFlag;


int NumberImage[NUMBER_IMAGE_MAX];        // 数字用画像

```

```
/******
```

```
* プロトタイプ宣言
```

```
*****/
```

```
/******
```

```
* ゲームメイン画面：初期化处理
```

```
* 引 数：なし
```

```
* 戻り値：エラー情報
```

```
*****/
```

```
int GameMainScene_Initialize(void)
```

```
{
```

```
    int ret = 0;
```

```
    int i;
```

```
    // 画像読み込み
```

```
    LoadDivGraph("images/number.png", NUMBER_IMAGE_MAX,
```

```
NUMBER_IMAGE_MAX, 1, 60, 120, NumberImage);
```

```
    // ステージ機能初期化
```

```
    ret = StageInitialize();
```

```
    // エラーチェック
```

```
    for (i = 0; i < NUMBER_IMAGE_MAX; i++)
```

```
    {
```

```
        if (NumberImage[i] == D_ERROR)
```

```
        {
```

```
            ret = D_ERROR;
```

```

        break;
    }
}

// ゲームプレイが初回かどうか？
if (GameCount == 0)
{
    GameScore = 0;                // スコアの初期化
    GameLevel = 1;                // ゲームレベルの初期化
    Set_StageMission(3);          // ミッションの初期化
    GameCount++;                  // 次回の設定
}
else
{
    GameLevel++;                  // ゲームレベルの更新
    Set_StageMission(3);          // ミッションを増やす
}

GameTime = TIMELIMIT;            // 制限時間の初期化

return ret;
}

```

```

/*****

```

```

* ゲームメイン画面：更新処理

```

```

* 引 数：なし

```

```

* 戻り値：なし

```

```

*****/

```

```

void GameMainScene_Update(void)

```

```
{  
    switch (Get_StageState())  
    {  
        case 0:  
            SelectBlock();          //ブロックを選択する。  
            break;  
        case 2:  
            MoveBlock();           //ブロックを移動させる。  
            break;  
        case 3:  
            CheckBlock();          // ブロックの確認  
            break;  
        case 4:  
            CheckClear();          // クリアチェック  
            break;  
        default:  
            break;  
    }  
}
```

// 制限時間の更新

GameTime--;

// 制限時間がなくなったら、ゲームオーバーに遷移する

if (GameTime < 0)

```
{  
    Change_Scene(E_GAME_OVER);  
}
```

```

// ミッションを達成したら、ゲームクリアに遷移する。
if (Get_StageClearFlag())
{
    Change_Scene(E_GAME_CLEAR);
}

}

/*****
* ゲームメイン画面：描画処理
* 引 数：なし
* 戻り値：なし
*****/
void GameMainScene_Draw(void)
{
    int PosX = 600;
    int tmp_level = GameLevel;
    int tmp_score = Get_StageScore();

    // ステージを描画
    StageDraw();

    // フェードアウト状態か？
    if (Get_StageState() == 1)
    {
        FadeOutBlock();          //フェードアウトする。
    }

```

```
//レベルを描画
```

```
do {
```

```
    DrawRotaGraph(PosX, 80, 0.5f, 0, NumberImage[tmp_level % 10], TRUE);
```

```
    tmp_level /= 10;
```

```
    PosX -= 30;
```

```
} while (tmp_level > 0);
```

```
//スコアを描画
```

```
PosX = 620;
```

```
do {
```

```
    DrawRotaGraph(PosX, 160, 0.3f, 0, NumberImage[tmp_score % 10],
```

```
TRUE);
```

```
    tmp_score /= 10;
```

```
    PosX -= 20;
```

```
} while (tmp_score > 0);
```

```
// 制限時間の描画
```

```
DrawBox(491, 469, 509, 469 - GameTime / 60 * 2, 0x0033ff, TRUE);
```

```
}
```