

```

//-----
// インベーダーゲーム
//-----
// EnemyShot.cpp
//
#include <DxLib.h>
#include "main.h"
#include "EnemyShot.h"
#include "Enemy.h"
#include "Player.h"

//-----
// 変数宣言
//-----
// 敵の弾
int enemyBulletImage; // 敵弾の画像ハンドル番号
dispData enemyBulletArray[ENEMY_DISP_YNUM][ENEMY_DISP_XNUM];

/**
*-----
* 敵弾の起動時のみの初期化処理
* Input:
*  無し
* Output:
*   true = 正常終了 / false = エラー終了
*-----
*/
bool EnemyShotSysInit(void)
{
    enemyBulletImage = LoadGraph("image/enemyBullet.png");
    if (enemyBulletImage == -1) return false;

    return true;
}

void EnemyShotInitProc(void)
{
    for (int yy = 0; yy < ENEMY_DISP_YNUM; yy++) {
        for (int xx = 0; xx < ENEMY_DISP_XNUM; xx++) {
            enemyBulletArray[yy][xx].pos.x = 0;
            enemyBulletArray[yy][xx].pos.y = 0;
            enemyBulletArray[yy][xx].flag = false;
        }
    }
}

/**
*-----
* 敵弾の発射処理
* Input:
*  無し
* Output:
*  無し
*-----
*/
void EnemyBulletShotProc(void)
{
    // 敵キャラの存在を調べて、発射処理を行う
    for (int xx = 0; xx < ENEMY_DISP_XNUM; xx++) {
        for (int yy = ENEMY_DISP_YNUM - 1; yy >= 0; yy--) {
            if (enemyFlgArray[yy][xx] == true) {
                // 存在している敵キャラが見つかった
                if (enemyBulletArray[yy][xx].flag == false) {
                    // 弾を撃てる状態
                    int rr = GetRand(ENEMY_BULLET_RAND_MAX);
                    if (rr < ENEMY_BULLET_SHOT_RAND_VALUE) {
                        enemyBulletArray[yy][xx].pos.x = enemyPosXArray[yy][xx] + ((ENEMY_SIZE_WID / 2) - (ENEMY_BULLET_SIZE_WID / 2));
                        enemyBulletArray[yy][xx].pos.y = enemyPosYArray[yy][xx] + ENEMY_SIZE_HIG;
                        enemyBulletArray[yy][xx].flag = true; // 弾を発射状態にする
                    }
                }
                // 弾を撃てる敵キャラがいたので、次の列に移行する
                break;
            }
        }
    }
}

```

```

/**
 *-----
 * 敵弾の移動処理
 * Input:
 *  無し
 * Output:
 *  無し
 *-----
 */
void EnemyBulletMove(void)
{
    for (int xx = 0; xx < ENEMY_DISP_XNUM; xx++) {
        for (int yy = ENEMY_DISP_YNUM - 1; yy >= 0; yy--) {
            if (enemyBulletArray[yy][xx].flag == true) {
                // 存在している敵弾のみ移動させる
                enemyBulletArray[yy][xx].pos.y += ENEMY_BULLET_MOVE; // 弾のY座標を移動量分加算する
                if (enemyBulletArray[yy][xx].pos.y > WINDOW_SIZE_HIG) {
                    // 敵弾がウィンドウの外に出てしまった
                    enemyBulletArray[yy][xx].flag = false; // 状態を非発射にする
                }
                // 同じ列でも、発射した敵がやられて、その上の敵が発射した弾が
                // 存在する可能性があるので、ここで break はしない
            }
        }
    }
}

/**
 *-----
 * 敵弾の描画処理
 * Input:
 *  無し
 * Output:
 *  無し
 *-----
 */
void EnemyBulletDraw(void)
{
    for (int xx = 0; xx < ENEMY_DISP_XNUM; xx++) {
        for (int yy = ENEMY_DISP_YNUM - 1; yy >= 0; yy--) {
            if (enemyBulletArray[yy][xx].flag == true) {
                // 存在している敵弾のみ描画する
                int posx = enemyBulletArray[yy][xx].pos.x;
                int posy = enemyBulletArray[yy][xx].pos.y;
                DrawGraph(posx, posy, enemyBulletImage, true);
                // 同じ列でも、発射した敵がやられて、その上の敵が発射した弾が
                // 存在する可能性があるので、ここで break はしない
            }
        }
    }
}

/**
 *-----
 * 敵弾とプレイヤーの当たり判定
 * Input:
 *  無し
 * Output:
 *   true = 命中 / false = はずれ
 *-----
 */
bool EnemyBulletHitCheck(void)
{
    bool bHit = false;
    for (int xx = 0; xx < ENEMY_DISP_XNUM; xx++) {
        for (int yy = ENEMY_DISP_YNUM - 1; yy >= 0; yy--) {
            if (enemyBulletArray[yy][xx].flag == true) {
                // 存在している敵弾の判定を行う
                int posx = enemyBulletArray[yy][xx].pos.x;
                int posy = enemyBulletArray[yy][xx].pos.y;
                if ((playerPosY < posy + ENEMY_BULLET_SIZE_HIG) && // ①プレイヤーの上 < 敵弾の下
                    (playerPosY + PLAYER_SIZE_HIG > posy) && // ②プレイヤーの下 > 敵弾の上
                    (playerPosX < posx + ENEMY_BULLET_SIZE_WID) && // ③プレイヤーの左 < 敵弾の右
                    (playerPosX + PLAYER_SIZE_WID > posx) && // ④プレイヤーの右 > 敵弾の左
                ) {
                    // プレイヤーに命中
                    enemyBulletArray[yy][xx].flag = false; // 敵弾を消す
                    playerFlg = false; // プレイヤーを消す
                    bHit = true;
                    break;
                }
            }
        }
    }
    if (bHit == true) break;
}
return bHit;
}

```

```
/**
 *-----
 * 敵弾の解放処理
 * Input:
 *  無し
 * Output:
 *   true = 正常終了 / false = エラー終了
 *-----
 */
bool EnemyBulletReleaseProc(void)
{
    if (DeleteGraph(enemyBulletImage) == -1) return false;

    return true;
}
```