

プログラマー勉強会 第一回

1. テンプレートを用意する

どのゲームでもだいたい同じなので、深く考えずに「こういうもんなんだな」としておく

ここから色々と書き加えていく

2. プレイヤー（画像）を表示してみる

- ・座標を格納するための変数を用意する

変数の宣言は、必ずメインループ前で行う

```
int i, j;
```

```
//プレイヤーの情報いろいろ
```

```
int player_x, player_y, player_life, player_xsize, player_ysize;
```

```
while (...
```

- ・画像を表示させるための変数を用意する

```
int i, j;
```

```
//プレイヤーの情報いろいろ
```

```
int player_x, player_y, player_life, player_xsize, player_ysize;
```

```
int player_g = LoadGraph("Data/player.bmp");
```

```
while (...
```

画像を読み込んで表示するためには、`int` 型の整数を使う

詳しく説明するとややこしくなるため、今はこういうものだと覚えておく

`LoadGraph` 関数：引数で指定された画像を読み込む

- ・画像を表示するために、プレイヤーの座標を設定する
タイトル画面からメイン画面に移行したときに設定されるようにする

```
//-----ゲームの処理をする-----  
//変数「mode」が0ならば、タイトル画面の処理をする  
if (mode == 0) {  
    if (key_z == 1) {          //Zキーが押されたら画面をタイトルからメインへ  
        mode = 1;  
        //ゲームに必要な情報を初期化していく  
        player_x = 200;  
        player_y = 420;  
        player_life = 5;  
        player_xsize = 40;  
        player_ysize = 40;  
    }  
}
```

ここに書けば、Zキーが押されてメイン画面に移行したときに実行されるようになる

- ・いよいよ画像を表示する

```
//-----ゲームの描画をする-----  
//変数「mode」が0ならば、タイトル画面の描画をする  
if (mode == 0) {  
    //文字を描画する  
    DrawString(20, 20, "タイトル画面", GetColor(255, 255, 255));  
}  
//変数「mode」が1ならば、メイン画面の描画をする  
else if (mode == 1) {  
    DrawGraph(player_x, player_y, player_g, TRUE);  
  
    DrawBox(0, 0, 440, 480, GetColor(0, 0, 255), FALSE);  
}
```

DrawGraph 関数：指定された x, y 座標に、指定された画像を描画する

最後の「TRUE」は、黒色を透明にするか否かの指定。今は「TRUE」なので、画像の黒色の部分は透明になる。「FALSE」にすると、そのまま黒色で表示される。

- プレイヤーを動かす

最初に、移動するためのキーを管理する変数を作る

Z キーの変数を参考に、十字キーの上下左右の変数を作る

```
int key_z = 0, key_up = 0, key_down = 0, key_left = 0, key_right = 0;
/* キーの入力状況を入れる。
   押している間は値が1ずつ増えていく。離すと0になる。
```

...

```
while (ProcessMessage() == 0 && CheckHitKey(KEY_INPUT_ESCAPE) == 0) {
    ClearDrawScreen();
    //キーの入力状況をチェックする
    //Zキーが押されていたら「key_z」を加算、そうでないなら0を代入
    if (CheckHitKey(KEY_INPUT_Z) == 1) key_z++;
    else key_z = 0;
    if (CheckHitKey(KEY_INPUT_UP) == 1) key_up++;
    else key_up = 0;
    if (CheckHitKey(KEY_INPUT_DOWN) == 1) key_down++;
    else key_down = 0;
    if (CheckHitKey(KEY_INPUT_LEFT) == 1) key_left++;
    else key_left = 0;
    if (CheckHitKey(KEY_INPUT_RIGHT) == 1) key_right++;
    else key_right = 0;
```

...

次に、キーの入力状態に合わせてプレイヤーの座標を変更する
ポイント：X座標とY座標の向きに注意する

```
//-----ゲームの処理をする-----  
//変数「mode」が0ならば、タイトル画面の処理をする  
if (mode == 0) {  
    ...  
}  
//変数「mode」が1ならば、メイン画面の処理をする  
else if (mode == 1) {  
    //キーが押されていたらプレイヤーを移動させる  
    if (key_up > 0)        player_y -= 4;  
    if (key_down > 0)      player_y += 4;  
    if (key_left > 0)      player_x -= 4;  
    if (key_right > 0)     player_x += 4;  
    //プレイヤーが画面外に出ないようにする  
    if (player_x < 0)       player_x = 0;  
    if (player_x + 40 > 440) player_x = 440 - 40;  
    if (player_y < 0)       player_y = 0;  
    if (player_y + 40 > 480) player_y = 480 - 40;  
}
```

3. 弾丸を打てるようにする

- 一度に表示される弾丸の最大数を決める
→ 弾丸を管理する変数の大きさを決める

```
#include "DxLib.h"  
  
#define WINDOW_XSIZE 640  
#define WINDOW_YSIZE 480  
  
#define MAX_SHOT 30
```

- ・弾丸に関するいろいろな変数を宣言する

```
//プレイヤーの情報いろいろ
int player_x, player_y, player_life, player_xsize, player_ysize;
int player_g = LoadGraph("Data/player.bmp");
int shot_trigger;
//弾丸の情報いろいろ
int shot_x[MAX_SHOT], shot_y[MAX_SHOT], shot_life[MAX_SHOT],
shot_xsize[MAX_SHOT], shot_ysize[MAX_SHOT];
int shot_g = LoadGraph("Data/shot.bmp");
```

- ・弾丸のデータを全て初期化する

タイトル画面からメイン画面に移行する時に、弾丸のデータを初期化する

```
//-----ゲームの処理をする-----
//変数「mode」が0ならば、タイトル画面の処理をする
if (mode == 0) {
    if (key_z == 1) {          //Zキーが押されたら画面をタイトルからメインへ
        mode = 1;
        //ゲームに必要な情報を初期化していく
        player_x = 200;
        player_y = 420;
        player_life = 5;
        player_xsize = 40;
        player_ysize = 40;
        shot_trigger = 0;

        for (i = 0; i < MAX_SHOT; i++) {
            shot_x[i] = 0;
            shot_y[i] = 0;
            shot_life[i] = 0;
            shot_xsize[i] = 0;
            shot_ysize[i] = 0;
        }
    }
}
```

- ・ Z キーを押したら弾丸が設置される

```
//-----ゲームの処理をする-----
//変数「mode」が0ならば、タイトル画面の処理をする
if (mode == 0) {
    ...
}
//変数「mode」が1ならば、メイン画面の処理をする
else if (mode == 1) {
    //キーが押されていたらプレイヤーを移動させる
    if (key_up > 0) player_y -= 4;
    if (key_down > 0) player_y += 4;
    if (key_left > 0) player_x -= 4;
    if (key_right > 0) player_x += 4;
    //プレイヤーが画面外に出ないようにする
    if (player_x < 0) player_x = 0;
    if (player_x + 40 > 440) player_x = 440 - 40;
    if (player_y < 0) player_y = 0;
    if (player_y + 40 > 480) player_y = 480 - 40;

    shot_trigger--;
    //Zキーが押されていて、かつトリガーが0以下だったら弾丸を発射する
    if (key_z > 0 && shot_trigger <= 0) {
        //使用されていない弾丸を探し、見つけたら情報をセットして抜け出す
        for (i = 0; i < MAX_SHOT; i++) {
            if (shot_life[i] == 0) {
                shot_life[i] = 1;
                shot_xsize[i] = 20;
                shot_ysize[i] = 20;
                shot_x[i] =
                    player_x + player_xsize / 2 - shot_xsize[i] / 2;
                shot_y[i] = player_y;

                //トリガーを設定
                shot_trigger = 10;
                break; //ループから脱出
            }
        }
    }
}
}
```

トリガー：いわゆるクールタイム。弾丸を発射したときにクールタイムがないと、Z キーを押したときに一気に何個も発射されてしまう

- ・ 設置した弾丸を動かす

```

//-----ゲームの処理をする-----
//変数「mode」が0ならば、タイトル画面の処理をする
if (mode == 0) {
    ...
}
//変数「mode」が1ならば、メイン画面の処理をする
else if (mode == 1) {
    //キーが押されていたらプレイヤーを移動させる
    if (key_up > 0) player_y -= 4;
    if (key_down > 0) player_y += 4;
    if (key_left > 0) player_x -= 4;
    if (key_right > 0) player_x += 4;
    //プレイヤーが画面外に出ないようにする
    if (player_x < 0) player_x = 0;
    if (player_x + 40 > 440) player_x = 440 - 40;
    if (player_y < 0) player_y = 0;
    if (player_y + 40 > 480) player_y = 480 - 40;

    shot_trigger--;
    //Zキーが押されていて、かつトリガーが0以下だったら弾丸を発射する
    if (key_z > 0 && shot_trigger <= 0) {
        //使用されていない弾丸を探し、見つけたら情報をセットして抜け出す
        for (i = 0; i < MAX_SHOT; i++) {
            if (shot_life[i] == 0) {
                shot_life[i] = 1;
                shot_xsize[i] = 20;
                shot_ysize[i] = 20;
                shot_x[i] =
                    player_x + player_xsize / 2 - shot_xsize[i] / 2;
                shot_y[i] = player_y;

                //トリガーを設定
                shot_trigger = 10;
                break; //ループから脱出
            }
        }
    }
    //登場している弾丸を動かす
    for (i = 0; i < MAX_SHOT; i++) {
        if (shot_life[i] > 0) {
            shot_y[i] -= 10;
            //弾丸が画面外に出たら削除する
            if (shot_y[i] < 0 - shot_ysize[i])
                shot_life[i] = 0;
        }
    }
}
}

```

- ・ 登場中の弾丸を表示する

```
//-----ゲームの描画をする-----  
//変数「mode」が0ならば、タイトル画面の描画をする  
if (mode == 0) {  
    //文字を描画する  
    DrawString(20, 20, "タイトル画面", GetColor(255, 255, 255));  
}  
//変数「mode」が1ならば、メイン画面の描画をする  
else if (mode == 1) {  
    for (i = 0; i < MAX_SHOT; i++) {  
        if (shot_life[i] > 0) {  
            DrawGraph(shot_x[i], shot_y[i], shot_g, TRUE);  
        }  
    }  
    DrawGraph(player_x, player_y, player_g, TRUE);  
    DrawBox(0, 0, 440, 480, GetColor(0, 0, 255), FALSE);  
}
```

Z キーを押すと弾丸が発射されるようになった・・・はず