

プログラマー勉強会 第三回

1. プレイヤーの体力を表示する

まずは画像を読み込む

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

体力の分だけ、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) {
    ...
}
//変数「mode」が1ならば、メイン画面の描画をする
else if (mode == 1) {
    ...
    DrawBox(0, 0, 440, 480, GetColor(0, 0, 255), FALSE);
    DrawString(460, 100, "life", GetColor(255, 255, 255));
    for (i = 0; i < player_life; i++) {
        DrawGraph(480 + i * 25, 120, plife_g, TRUE);
    }
}
```

2. プレイヤーと敵の当たり判定をつける

弾丸と敵の時と条件は同じ

```
//-----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) {
    ...
    //敵と弾丸の当たり判定
    for (i = 0; i < MAX_SHOT; i++) {
        ...
    }
    //敵とプレイヤーの当たり判定
    for (i = 0; i < MAX_ENEMY; i++) {
        if (enemy_life[i] > 0) {
            if (
enemy_x[i] <= player_x + player_xsize && enemy_x[i] + enemy_xsize[i] >= player_x) {
                if (
enemy_y[i] <= player_y + player_ysize && enemy_y[i] + enemy_ysize[i] >= player_y) {
                    player_life--;
                }
            }
        }
    }
}
}
```

プレイヤーが敵にぶつくと、プレイヤーの体力が減るようになった
でも・・・

3. プレイヤーに無敵時間をつける

```
//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_safetime; //ダメージを受けないセーフタイム
int player_g = LoadGraph("Data/player.bmp");
```

宣言したら、いつもの初期化

```
//-----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;
        player_safetime = 0;
        shot_trigger = 0;
        ...
    }
}
```

セーフタイムが1以上なら、先ほど書いた当たり判定をスキップするようにする

```
//-----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) {
    ...
}
//変数「mode」が1ならば、メイン画面の処理をする
else if (mode == 1) {
    ...
    //敵と弾丸の当たり判定
    for (i = 0; i < MAX_SHOT; i++) {
        ...
    }

    //プレイヤーのセーフタイムを1減らす
    if (player_safetime > 0)
        player_safetime--;
    //敵とプレイヤーの当たり判定
    if (player_safetime == 0) {
        for (i = 0; i < MAX_ENEMY; i++) {
            if (enemy_life[i] > 0) {
                if (...) {
                    if (...) {
                        player_life--;
                        player_safetime = 60;
                    }
                }
            }
        }
    }
}
}
```

あたり判定全体のインデントをごそっと下げる必要がある

→ visual studio が勝手にやってくれる

実行すると、ちゃんと体力が1個ずつ減るようになった

4. 無敵時間中、プレイヤーを点滅させるようにする

無敵時間は、目に見えたほうがいい

```
//-----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) {  
    ...  
}  
//変数「mode」が1ならば、メイン画面の描画をする  
else if (mode == 1) {  
    ...  
    if (player_safetime % 2 == 0)  
        DrawGraph(player_x, player_y, player_g, TRUE);  
}
```

%演算子： $a \% b$ の場合、 a を b で割った時の余りを算出してくれる

例： $15 \% 6$

= 3

`player_safetime` を 2 で割った時の余りが 0 なら（つまり、`player_safetime` が偶数なら）、プレイヤーの画像を表示する

実行すると、無敵時間中はプレイヤーが点滅するようになった

5. スコアを実装する

いつものように宣言から

```
//敵の情報いろいろ
...
//ゲームの情報いろいろ
int score;
```

いつものように初期化

```
//-----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
        ...
        score = 0;
    }
}
```

敵を倒したらスコアを増やす

```
//-----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) {
    ...
    //敵と弾丸の当たり判定
    for (i = 0; i < MAX_SHOT; i++) {
        if (shot_life[i] > 0) {
            for (j = 0; j < MAX_ENEMY; j++) {
                if (enemy_life[j] > 0) {
                    if (...) {
                        if (...) {
                            //当たっていたら、弾丸を削除して敵の体力を1減らす
                            shot_life[i] = 0;
                            enemy_life[j]--;
                            //敵を倒したらスコアを増やす
                            if(enemy_life[j] == 0)
                                score += 100;
                        }
                    }
                }
            }
        }
    }
    ...
}
}
```

スコアを表示する

```
//-----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) {
    ...
}
//変数「mode」が1ならば、メイン画面の描画をする
else if (mode == 1) {
    ...
    DrawBox(0, 0, 440, 480, GetColor(0, 0, 255), FALSE);

    DrawString(460, 20, "score", GetColor(255, 255, 255));
    DrawFormatString(480, 40, GetColor(255, 255, 255), "%d", score);
    DrawString(460, 100, "life", GetColor(255, 255, 255));
    ...
}
```

6. ゲームオーバーをつくる

プレイヤーの体力が0になったら、mode をゲームオーバーにする

```
//-----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) {
    ...
}
//変数「mode」が1ならば、メイン画面の処理をする
else if (mode == 1) {
    ...
    //敵とプレイヤーの当たり判定
    if (player_safetime == 0) {
        ...
    }
    //プレイヤーの体力が0以下になったらゲームオーバー
    if (player_life <= 0)
        mode = 2;
}
}
```

実行すると、体力が0になった時に真っ暗な画面が表示される

ゲームオーバー画面からタイトルに戻す

```
//-----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) {
    ...
}
//変数「mode」が2ならば、ゲームオーバー画面の処理をする
else if (mode == 2) {
    //Zキーが押されたらタイトル画面に戻る
    if (key_z == 1)
        mode = 0;
}
}
```

ゲームオーバー画面をつくる

```
//-----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) {  
    ...  
}  
//変数「mode」が1ならば、メイン画面の描画をする  
else if (mode == 1) {  
    ...  
}  
//変数「mode」が2ならば、ゲームオーバー画面の描画をする  
else if (mode == 2) {  
    DrawString(20, 20, "Game Over", GetColor(255, 255, 255));  
    DrawFormatString(20, 100, GetColor(255, 255, 255), "Total Score : %d", score);  
}  
}
```