

## プレイヤーの攻撃

さて、横スクロールアクションの制作が始まりました。今回のゼミではプレイヤーの描画、そしてその移動をしました。そこで今回の追加資料ではその描画したプレイヤーにジャンプ以外のちょっとしたアクションを起こしたいと思います。

そのアクションとは攻撃、任意のボタンを押すとプレイヤーが攻撃、ここでは衝撃波つぼいのを目の前に出します。

### ◆ 攻撃

まずプログラム内にコメントアウトされているところがありますのでそのコメントアウトを外してください。

`move` 関数内のコメントアウトされている部分は「攻撃中に方向転換できないようにする」というプログラムが入ります。

```
void attack(void){
    GetHitKeyStateAll(keyState);
    if(keyState[KEY_INPUT_S]){
        if( player.attack == 0 ){
            player.attack = 1;
        }
    }
    else{
        【穴埋め】
    }
}
```

攻撃をする関数、`attack` 関数を追加しましょう。

この関数では `S` のボタンを押すとプレイヤーが攻撃状態になります。

ストラクトの部分にも書いてある通り、`player.attack` が攻撃中か否かを示す関数であり、攻撃をしていない時は `0` が代入され、攻撃中の時は `1` が代入されます。

```

if(player.attack){
    if(player.dire == 0){                //主人公攻撃書くよ
        DrawGraph(【穴埋め】,player.y,img.migiattack,TRUE);
    }
    else{
        DrawGraph(【穴埋め】,player.y,img.hidariattack,TRUE);
    }
}

```

プレイヤーの攻撃（衝撃波っぽい）を draw 関数にて描画しましょう。

player.attack が 1 の時＝プレイヤーが攻撃中の時描画されます。ここではその描画する x 座標が穴埋めになっています。



攻撃中の描画は上のようになります。プレイヤー自身の座標 (player.x) と比べ、衝撃波の座標はどのあたりになるか考えてみましょう。