

ゲームプログラミング講座 ループ編

ループというのは同じ処理を繰り返すこと。(例:for 文や while 文)

はじめに、**ゲームはループで成り立っている**。(説明は後ほど)

ゲームの要素

| 要素名 | 内容 |
|-----|--|
| 初期化 | ウィンドウの生成、ファイルからのデータや画像の読み込み、数値の初期化 |
| 入力 | キーボード、マウス、パッド(ゲームのコントローラー)等からの入力の状態を得る |
| 処理 | 入力に対するキャラやカーソルの移動、敵の移動処理など |
| 出力 | 画面の描画、音楽の再生 |
| 終了 | セーブデータの保存など |

ゲームはこれらの要素から構成されている。

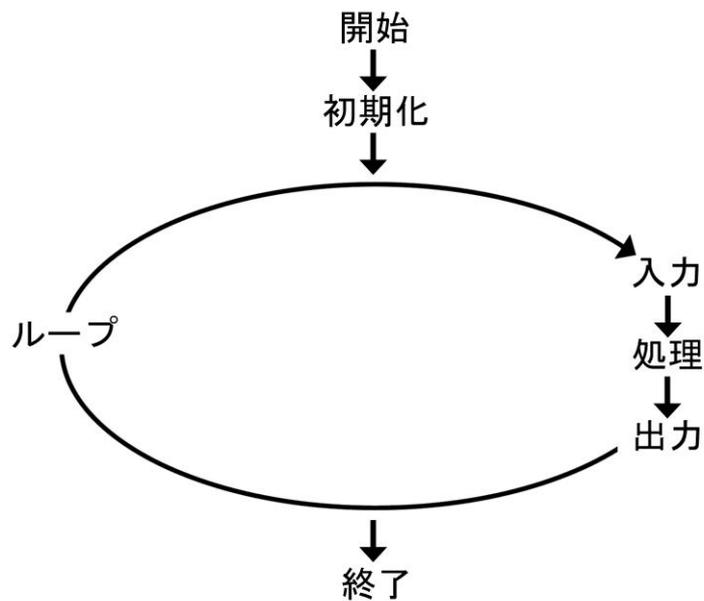


図1 ゲームの流れ

```
main()
{
    /// 開始
    初期化(データの読み込み等)
    /// ゲームループの開始(無限ループ)
    while(1)
    {
        入力(キーボード、パッド)
        ゲーム処理
        出力(描画、音)
    }
    終了処理
    /// 終了
}
```

Q. ゲームと無限ループの関係について

家庭用ゲーム(ゲームソフト)は、そのゲームで終了コマンドを行ってもタイトルに戻ったりするだけで電源が切れることはない。つまりハードの電源を切らない限り動き続ける。 → 無限ループ

Q. 入力を調べる時もループは回っているか？

A. 入力などのイベントが起こった時に処理をするのではなく、プレイヤーの操作とは無関係にゲームループは常に回っている。だから入力以外の処理はずっと行われていて、敵が動いたり弾を発射してきたりする。

詳しくは次のページへ

次の二つのプログラムを比較してみよう。

ループ内で処理が止まっている場合 <例: 改行するまで入力の確認を続けるプログラム>

```
main()
{
    char c;

    while(1)
    {
        scanf("%c", &c);    //ここで処理はいったん止まっている

        if( c == '\n' ) break;    //変数 c に改行が入るとループを抜ける
    }
}
```

ゲームループの場合

```
main()
{
    int player = 0, enemy = 0, bullet = 0;

    while(1)
    {
        //player の値はキーが入力されていると1、2、3・・・と増加する
        if(キーが入力されている) player++;

        //enemy、bullet の値は入力を行わなくても1、2、3・・・と増加する
        //→入力しなくてもゲームは動き続ける(処理は止まらない)
        enemy++;
        bullet++;
    }
}
```

ゲームループの中で入力されているかどうかを毎ループ調べている。

ゲームループの参考例: RPGの戦闘シーン

(ファイナルファンタジーとドラゴンクエストの戦闘シーンは、ゲームループの点では同じであること)

もっと具体的にシューティングゲームを例にとって説明してみよう。

<シューティングゲームのループ処理>

```
main()
{
    初期化(自機、敵、弾、ステージなど)

    /// ゲームループ
    while(1)
    {
        (1) 入力
        (2) 自機移動
        (3) 自機からの弾の発射
        (4) 敵の移動、弾の発射
        (5) 弾の移動(自機、敵)
        (6) 自機の弾が敵に当たっているか調べる
        (7) 敵の弾が自機に当たっているか調べる
        (8) 画面に出力、音を鳴らす
    }
    終了(スコアの保存など)
}
```

基本的に(1)~(8)の処理を繰り返す。

<発展>

ゲームの状態にはゲームメイン、タイトル、メニュー、オプション、ポーズなどがあり、それぞれに対する処理が必要。

ゲームの状態を表す変数を作って処理を分けると良い。

FPSとは

FPS:(Frame Per Second:単位秒あたりのフレーム数)

つまり、1秒あたりにフレームが呼ばれる回数のことをいう。1フレームというのは1ループのこと。

FPSで管理するのはフレーム数であり、ゲームでは約 60Hz(1秒間に60回)の処理を行う。
これはディスプレイの更新が1秒間におよそ60回であることからである(これをリフレッシュレートという)。

先ほどのゲームループの例では1秒間キーを押していた場合、player の値は 60 になる。
また、多くの処理をしたり、たくさん描画したりすると、処理が重くなり、FPSの値が小さくなる。
これを処理落ちという。

ゲームでフレーム管理を行わないと・・・

パソコンの処理速度によってゲーム自体の速さが変わってしまう。

つまり、環境によって全く違った難易度のゲームになってしまう。

(処理速度が速い → ゲームスピードが速くなる)

対策として、FPSの上限を固定する方法がある。