

この時間のねらい：① 処理の種類を知ろう。

② 今回の内容の処理（反復処理）について学ぼう。

皆さんが遊んでいるゲームには何か処理が行われているのでしょうか？例えば、素材を集めたいとき、また、特定の場所に出てくるモンスターを捕まえたいときには、周回プレイを出るまで行うというのが、やり方の一つです。このように、プログラミングとして何度も同じことを続けさせる処理を **反復処理** 又は **ループ処理** と言います。

他に、そこそこ前の話ですが、「五等分の花嫁」という漫画が完結しました。最終的には○番目の娘と結婚しました。(ネタバレ回避のため隠しています。) この結末には、なかなか炎上が起こったようです。話を戻して、この内容には決定的なフラグがあり、その条件を満たしたことで、その結末になっています。このように、プログラミングなどでも、条件をつけて、それが満たされたかどうかを判断して、処理を変更することを 条件分岐 と言います。

最後に、今までも自然にやってきていたことですが、例えば、上から順番に三角形、丸、四角形を書くプログラムを作ったとします。すると、一番最後に書いたものが一番上になるようになっています。(これは絶対です。)

このように、プログラムは上から一つ一つ順番に処理するということを 順次処理 もしくは 逐次処理 と言います。

逐次処理に関しては、基本的にやりながらで分かっていきますから、今回は取り扱いません。ほかの2つの処理を今回と次の内容で学びます。

それでは 反復処理、ループ処理 の話に移りましょう。例えば、皆さんの使用している表計算ソフトである Microsoft Office Excel には数字や曜日などを連続してつけてくれる機能がついています。これを オートフィル と言います。

この機能は何も連続性の無い言葉や文字の場合であれば、そのまま同じものを別セルに入れていきます。これは言い換えれば同じ文字列を 反復して 表示させることと同じです。要は、 **反復処理** の一例といえるということになります。

それでは、実際に使う方法を見てみましょう。右のやり方は java での宣言方法と全く一緒です。(教科書 p.61 参照)

```
for(int i = 1; i < n+1回目; i++){
    くり返す処理の内容
}
```

さて、これはどんなことをしているかという引数のあるはずの括弧内には基本的な条件が書かれています。この条件の書き方が面倒ですが、繰り返す回数を表す変数 i を宣言し、それを何回行うかは不等号を用いて表します。増やし方は基本的には 1 ずつのため上のようにしておけば大丈夫です。

<増やし方について>

先ほどの for 文での増やし方は他にもあり、例えば、i += 2 と記述すれば 2 ずつ増加させられたり、i=i+2 でも 2 ずつ増加になります。

また、もう一つ while 関数も反復処理の関数の一つです。(教科書 p.65 参照)

```
while(式){
    くり返す処理の内容
}
```

この式の使いどころは 事前に 宣言してある変数について条件を付けることができます。

ただし、この条件が偽にならない場合には、この関数に書かれている処理はずっと終わることができません。このような処理を 無限ループ と言います。そのため、デメリットとしては、場合により、この状態になってしまい、ほかの処理を行うことが難しい状態になるということが言えます。これらの文を利用することでできることは、同じものを複数出すことや、まったく同じものをランダムな場所に表示するなどができます。

基本的には横に並べることが目的の繰り返しですが、2 つ重ねて利用することで、縦横の両方向への並びも制御させることができます。

for 関数の重ね方（以下に手書きしましょう）