

----- 今日の目標 -----

教科書第6章「繰り返し」の6.1~6.3を学ぶ。 次の問に答えられるようにする。

- for文で単純な繰り返しを書けるようになる。 その時変数の上に重ね書きをする手法を使えるようになる。
- do ~ while を用いた繰り返しを書けるようになる。

1. 簡単な繰り返し forループ

[繰り返しの例] 1からN (たとえば10とか) までの和を求める。つまり $S = 1 + 2 + 3 + 4 + \dots + 10$

```
int S = 0;
i が 1 から 10 まで順番に繰り返す {
    S = S + i;
    System.out.println(S);
}
```

このように繰り返し (ループ) の形にすると、次のように動作する。

- iが1のとき: $S = S + i = 0 + 1 = 1$
- iが2のとき: $S = S + i = 1 + 2 = 3$
- iが3のとき: $S = S + i = 3 + 3 = 6$
- (途中省略)
- iが10のとき: $S = S + i = 45 + 10 = 55$

[Javaでの書き方]

```
for (i=0; i<=10; i++) {
    ... ..
}
```

これを読み下すとすれば、

- 最初に (繰り返し (ループ) を始める前に) iを0にする (初期化)
- 繰り返し (ループ) を終わる条件は、 $i \leq 10$ とする つまり、iが10になるまで繰り返す (終了条件)
- iの値は、繰り返しを1回まわるごとに、i++する つまり、 $i = i + 1$ 、つまり、iを1増やす
- で、ループの中身 (本体、ボディ) は、{ } である (上記の例なら、 $S = S + i$)

```
for (int i=0; i<=10; i++){           for ( <初期化>; <終了条件>; <1回ずつ変化の式> ) {
    S = S + i;                       ループの本体 (ボディ);
}                                     }
```

[演習1] 教科書 p151 Sample2 を試してみよう

[演習2] Sample2 をコピー・改造してSample2-2とする。Sample2-2では、奇数 (1, 3, 5, ..., 9) だけを書くプログラムを作ろう

```
1です
3です
5です
7です
9です
```

〔演習3〕 Sample2-2をコピー・改造してSample2-3とする。Sample2-3では、1から10までの和をプリントするプログラムを作ろう

```
1
2
3
6
10
15 (以下略)
```

〔演習4〕 Sample2-3をコピー・改造してSample2-4とする。Sample2-4では、キーボードから整数Nを入力し、1からNまでの和をプリントするプログラムを作ろう。

〔演習5〕 Sample2-4をコピー・改造してSample2-5とする。Sample2-5では、キーボードから整数Nを入力し、(演習4とは逆順に) Nから1までの和をプリントするプログラムを作ろう。

```
10 ← 10を入力
19 ← 10+9
27 ← 19+8
34 ← 27+7
40 ← 34+6
(以下略)
```

〔演習6〕 演習4のSample2-3をコピー・改造してSample2-6とする。Sample2-6では、キーボードから整数Nを入力し、1からNまでの積 (=階乗!) を1つ1つ順番にプリントするプログラムを作ろう。

```
1
2 ← 1×2
6 ← 2×3
24 ← 6×4
120 ← 24×5
(以下略)
```

<問題>

- ① Nを20として、20までの階乗を計算してみよ。 マイナスになる? 合わない?
- ② 高校で習ったと思うが、「順列・組合せ」の中で階乗が出てくる。では、15枚のカードから5枚を取る取り方つまり組合せを計算するにはどうしたらよいか、考えてみよう。

$$nC_r = n! / (r! \times (n-r)!)$$

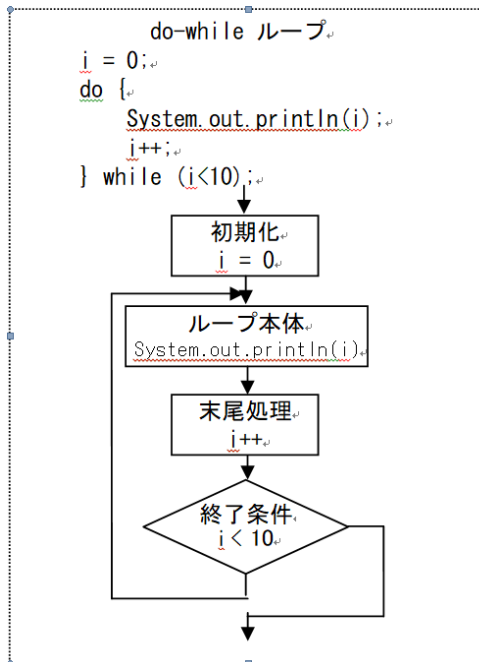
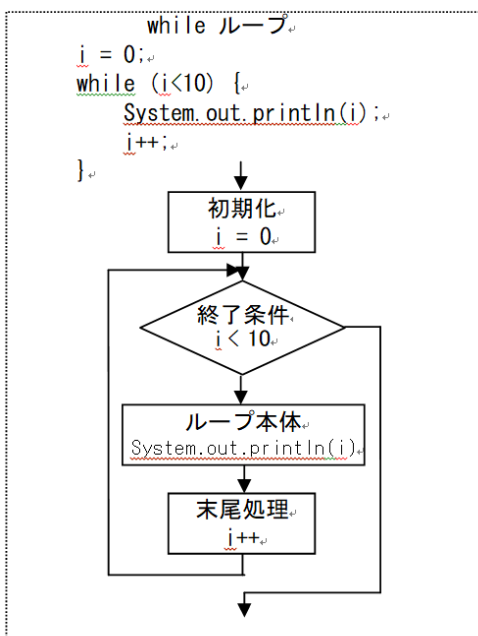
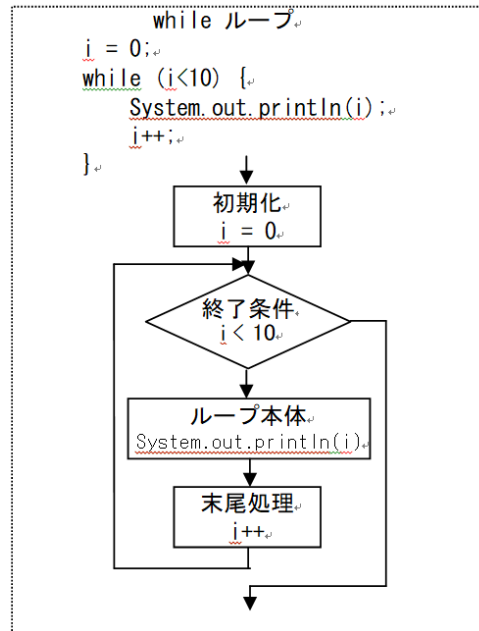
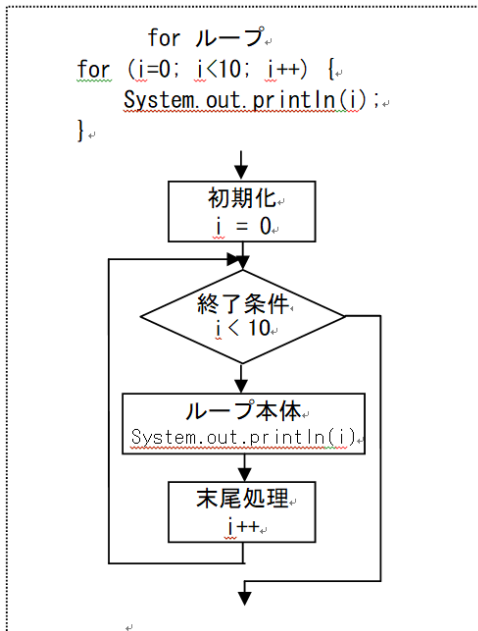
で、 $n = 15$ 、 $r = 5$ を考える。分子は15の階乗なのだが、どうやって計算しようか ... ?

<脱線>

```
for (i=0; i<5; i++) { ... }
```

としたときに、**i**の値をループ本体の中で変更してよいか?

2. もう1つのループ whileループ と do～whileループ (脱出条件判定の位置が違う)



<問い> どのようなときに、do-whileループが必要だろうか？ 下の例で、使う条件を考えてみよう。

```
do {
    キーボードから (ファイルから) データを入力。
    〃
    入ってきた入力を処理・出力。
    〃
} while (データが“終わり”でないなら);
```

[演習7] 教科書p 173の問題2を試してみよう。

キーボードから点数 (整数) を入力し、その数値が0だったら合計点を表示して終了する。0でなければ次を入力。

[演習8] 演習7の修正版 ⇒ 0を入力したら、(合計点ではなくて)、「平均点」を出して欲しい。