

プログラミング入門第5回

箕原辰夫
2005年秋学期

for文を使った繰り返し

- for文の書式

```
for (最初の式 ; 継続条件 ; 繰り返す式) {  
    繰り返したい内容  
}
```

- 意味

1. 最初の式が、まず実行される
2. **while**文と同じで継続条件を評価する
3. 繰り返したい内容を実行してから、繰り返す式を評価
4. 2のところに戻る

for文の実行の模式

- for文の各部分をABCDで表わす
 - **for** (A; B; C) { D }
- 実行は以下のようなになる
 - A → B → D → C → B → D → C → B → D → C → ...
→ B → D → C → B → 次へ
- 省略について
 - A, C, D…省略可能
 - B…省略可能だが、省略された場合はtrueだとみなす

ループ変数の初期値設定

- 最初に行なうところ、宣言文を書いても良い (Java/C++)
- そこで宣言された変数は、**for**文のブロックを抜けたら消滅する
- ループ変数を増分させたときの繰返し回数
の目安

`for ( int i=初期値 ; i < 上限値 ; i += 間隔 ) { }`

- このような場合の繰返しの回数
→ $(\text{上限値} - \text{初期値}) / \text{間隔}$ ※整数除算

for文を使ったさまざまな記述例

- 単純にループ変数を増分させる
- ループ変数を減らしていき、最終値をマイナスの値と比較
- ループ変数を掛け算で増やしていく
- 何もしないで繰り返す

総和を求める

- ループ変数の値の変化に注目
- 足し合わされる変数 sum の変化にも注目する



止まる設計

- 初期値と継続条件の設定の仕方による
 - 1 回も実行されない
 - 止まらなくなる
- 文法間違い
 - **for**の括弧の中をカンマで区切る→正解は
セミコロン
 - **for**の括弧の直後にセミコロンをいれる→
別ブロック

while文とfor文との互換性

- while文をfor文で書き直す

→ while (B) { D }

→ for (; B;) { D }

- for文をwhile文で書き直す

→ for (A; B; C) { D }

→ { A; while (B) { D C; } }

break文

- **break**文に出会うと、一番内側のブロックから脱出する
- ブロックのネストが深い場合は、脱出レベルに対してラベルをつける

continue文

- 繰返して、そのブロックの残された部分を飛ばして、次の繰返しをはじめる

return文

- メソッド自体を終了させる
- 呼出し側に値を返すこともできる（第 1 5 章）

論理値を保持する変数

- **boolean**型で変数を宣言する
 - **boolean** *flag*;
- 論理値を代入することができる。
 - *flag* = **true**; *flag* = **false**;
- 条件式の評価結果を代入することができる
 - *flag* = (*x* >= 100);
- **if**文の中で変数の値を使うことができる
 - **if** (*flag*) { }

do～while文

- 1回は実行してしまい、実行の最後で繰返すかどうかチェックする
 - ➔ **do** { 繰返すこと } **while** (継続条件);
- 最後にセミコロンが必要なので注意
- ときどき使う
 - ➔ **for**や**while**だと同じことを2箇所にかかなくちゃならない場合

switch文

- if文で定数と比較する場合に利用

```
switch (変数や式) {
```

```
case 定数値: 実行すること; break;
```

```
case 定数値: 実行すること; break;
```

```
default: 実行すること; break;
```

```
}
```

- **break**を書きわすれると次の**case**も実行されてしまう

if式

- **if**で、評価する値をどちらかに決定できる
(条件・論理式) ? 真の場合の値 : 偽の場合の値
- 式なので、代入文の中やメソッド呼出しのパラメータの中でも使える
 - ➔ `x = ( y >= 100 ) ? 10 : 20;`
 - ➔ `g.setColor( ( x > 100 ) ? Color.red : Color.blue );`

if式（続き）

- if式をネストさせることもできる

```
c = ( y >= 80 ) ? 'A' : ( y >= 60 ) ? 'B' :  
    ( y >= 40 ) ? 'C' : 'D' ;
```

- かなり多用するプログラマが多い
 - if文が省略できる、短く書ける
 - 使い過ぎると何をしているのかわからないので、ほどほどに

制御構文における変数の有効領域

- ブロック内で宣言された変数 (**for**文も含む)
 - ブロックが消滅すると、一緒に消滅する
- 繰返しのブロックの中で初期値代入されて宣言された変数
 - 毎回の繰返しで、初期値が代入される

課題6-6のバベッジの階差機関

- 2 次方程式に値を代入して計算値を求める
- 掛け算を足し算で計算していく方法

x :	1	2	3	4	5	6	7	8	9
x^2 :	1	4	9	16	25	36	49	64	81
第1階差:		3	5	7	9	11	13	15	17
第2階差:			2	2	2	2	2	2	2

今日の課題

- 宛先 : minohara@sfc.keio.ac.jp
- 件名 : java-3
- 期限 : 11月10日
- 課題 : 教科書の課題6-7
- 内容 : プログラムをコピー & ペースト、質問、授業の感想