プログラミング基礎

第2.5回

やり直しのif文，for文，while文

boolean型

- 普段あまり表に出てこないが，
  if文，for文，while文の主役
- boolean型：true，falseだけを記憶する型

例：

```
boolean x;
x = true; //trueは文字列ではないことに注意
x = false; //falseも同様
```

boolean型

- ではtrue，falseを計算するには？
  ⇒それが今まで使ってきた
  - 比較演算子
  - 論理演算子

比較演算子

- A < B：AがBより小さいのであればtrue，それ以外はfalse
- A <= B：AがB以下であればtrue，それ以外はfalse
- A > B：AがBより大きいのであればtrue，それ以外はfalse
- A >= B：AがB以上であればtrue，それ以外はfalse
- A == B：AとBが等しいのであればtrue，それ以外はfalse
- A != B：AがBが等しくないのであればtrue，それ以外はfalse

例：

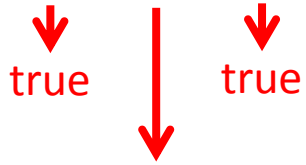
```
int x = 10; int y = 15;
boolean r;
r = x == 11;          //rはfalse
r = x <= y;           //rはtrue
r = x < 5;            //rはfalse
r = x != y;           //rはtrue
```

論理演算子

- A && B：AとBが共にtrueであればtrue，それ以外はfalse

- A || B：AとBが共にfalseであればfalse，それ以外はtrue

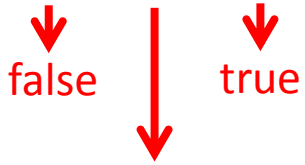
例：
```
int x = 10;
boolean r;
r = (x > 5) && (x < 15);
```



共にtrueなので
⇒rはtrue

日本語で表せば，
「xが5より大きい，かつ，
15より小さければtrue」

例：
```
int x = 10;
boolean r;
r = (x == 5) || (x == 10);
```



共にfalseではないので
⇒rはtrue

日本語で表せば，
「xが5，または，10
であればtrue」

if文

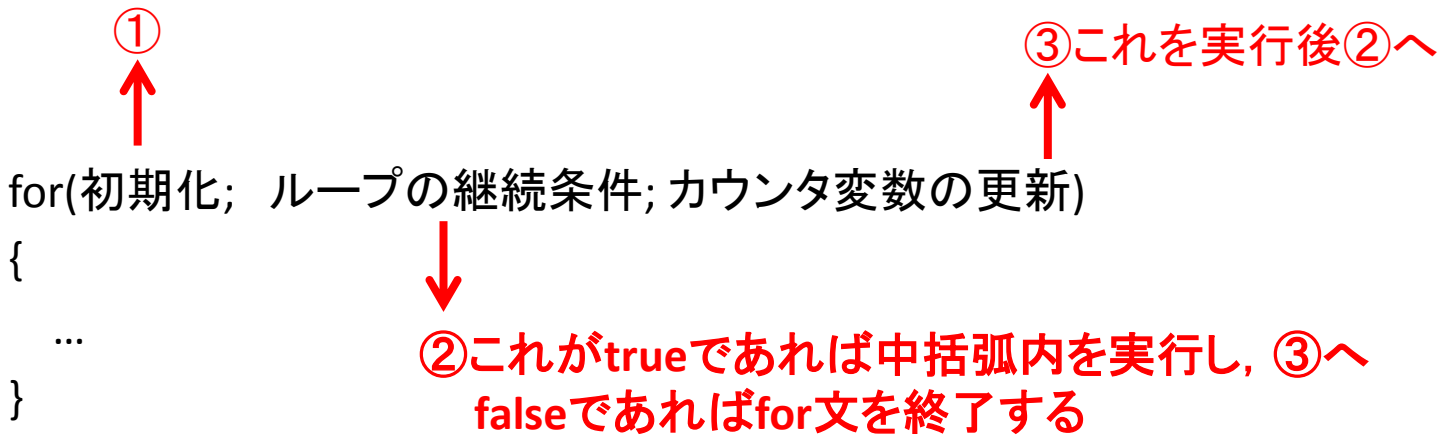
```
if (  )
{
    …
}
else
{
    …
}
```

括弧内がtrueであれば，

すぐに続く中括弧内を実行，

falseであればelseに続く中括弧内を実行

for文


①
③これを実行後②へ
for(初期化;　ループの継続条件; カウンタ変数の更新)
{
…
}
②これがtrueであれば中括弧内を実行し，③へ
falseであればfor文を終了する

while文

```
while(   )
{
    …
}
```

①括弧内がtrueであれば，

②すぐに続く中括弧内を実行
した後，①に戻る

他のboolean型の使いどころ

- 他にも以下のような二値の情報を記憶するために使用する
  - 特定の処理が終わったかどうか
  - ある機能が有効になっているか

⇒プログラミング用語で「**フラグ**」と呼ぶ

例:

```
boolean isCompleted;
…
isCompleted = ある処理;
…
if (isCompleted)
{
   //ある処理が終わっていたときの処理
}
```

補足

- double（float）型に記憶された値は
  浮動小数点数
  呼ばれる

- 例：12500⇒1.25×10^4
  上記のように，**小数点位置を動かして**記憶されることから

- （過去には，ゲームなど速さが求められるアプリケーションでは，int型などの整数型で小数を表現する**固定小数点**数という手法があったが，現在ではほとんど出番はない）