

# 計算機とアルゴリズム I 第 11 回

## 反復文 (for 文)

森 義之

岡山理科大学

# for 文

for 文は制御式の真偽値が真の間は繰り返し、偽になると終わる。

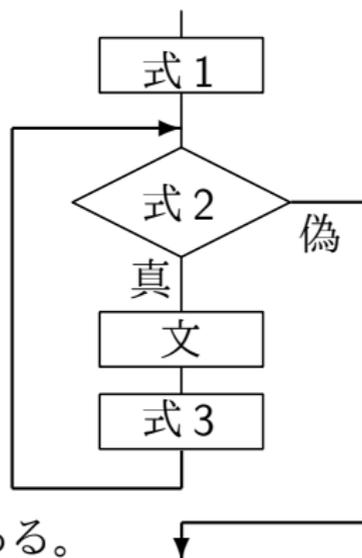
## 書式

```
for(式1 ; 式2 ; 式3 ) 文;
```

for 文の流れは以下のようなになる。

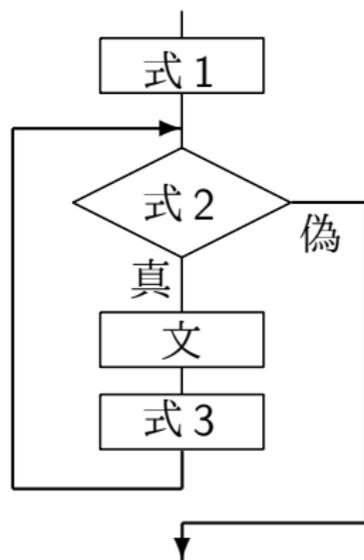
- (1) 式1 を実行する。
- (2) 式2 が真のときは文を実行する。  
偽なら (4) に行く。
- (3) 式3 を実行し、(2) に戻る。
- (4) for 文を終了する。

このとき、式1, 式2, 式3 は省略が可能である。



## 例 4.4.1. (for 文の例)

- 1) `for(i=1;i<=10;i++)`
- 2) `for(i=10;i>0;i--)`
- 3) `for(i=1;i<10;j+=2)`
- 4) `for(i=1;(i<10 && j<5);i++)`
- 5) `for(;i!=0;)`



良く使われる方法として、“ $n$ 回 文を実行する” という繰り返し (ループ) 処理がある。

## 例 4.4.2. (指定回数反復処理 1)

```
for(i=1;i<=3;i++){  
    printf("こんにちは ¥n");  
}  
printf("おわり ¥n");
```

実行結果

```
こんにちは  
こんにちは  
こんにちは  
おわり
```

## 例 4.4.3. (指定回数反復処理 2)

```
for(i=1;i<=3;i++)  
    printf("%d 回目のループ ¥n",i);  
printf("おわり ¥n");
```

実行結果

1 回目のループ  
2 回目のループ  
3 回目のループ  
おわり

## 課題 J.1.

以下を入力&実行し、結果を確認せよ。

```
#include<stdio.h>
int main(void){
    int i,a;
    printf("自然数を入力してください:");
    scanf("%d",&a);
    for(i=0;i<a;i++){
        printf("%d 回目のループ %n",i+1);
    }
    return 0;
}
```

※ scanf 文で入力する数値を変えて試してみる。

♠ 補足 再設定式 (式 3) には

$$i++, ++i, i+=1, i=i+1$$

のように 1 増加するようなもの以外の式も記述できる。例えば、

$$i--, i=i-1, i*=2, i/=2$$

のように 1 ずつ減少や 2 倍増, 半減なども可能である。さらに、式 1, 式 2 で書かれた変数以外で指定することも可能である。

# for 文

また、例 4.4.5. のように、for 文を抜けた後に継続条件で用いた変数の値にも注意すること。

for 文は  $i$  が 6 になった時点で終了しているので、その後使用する  $i$  の値は 6 である。

## 例 4.4.4. (指定回数反復処理 3)

```
pow=1;
for(i=1;i<=5;i++){
    pow*=2;
    printf("2 の%d 乗は%3d です。 ¥n",i,pow);
}
```

### 実行結果

```
2 の 1 乗は  2  です。
2 の 2 乗は  4  です。
2 の 3 乗は  8  です。
2 の 4 乗は 16  です。
2 の 5 乗は 32  です。
```

## 例 4.4.5. (指定回数反復処理 4)

2 の 5 乗を計算する (つもりで作った) プログラム

```
pow=1;
for(i=1;i<=5;i++)
    pow*=2;
printf("2 の%d 乗は%d です。¥n",i,pow);
```

実行結果

2 の 6 乗は 32 です。

(何かおかしい??)

## 課題 J.2.

以下のプログラムの.....部分を修正し、1から入力された自然数までの和を求めるプログラムを作成せよ。

```
#include<stdio.h>
int main(void){
    int i,a,sum=0;
    printf("自然数を入力してください:");
    scanf("%d",&a);
    .....
    .....
    .....
    printf("1から%dまでの自然数の和は%dです。¥n",a,sum);
    return 0;
}
```

※ ヒント : 例 4.7.1.

例 4.4.6. (i が 100 以下 かつ sum が 1000 以下で繰り返す)

```
sum=0;
for(i=1;(i<=100)&&(sum<=1000);i++)
    sum+=i;
```

例 4.4.7. (和が負でない場合に繰り返す for 文)

```
sum=0;
for(i=1;!(sum<0);i++)
    sum+=i;
```

このプログラムでは正の数を足し続けると、いつか負の数になることを意味している。

# for 文

☆ for 文のなかに for 文を入れることができる。

そのときは、ループの回数を表す変数 (式 2 で使用している変数) の使用に注意すること。

## 課題 J.3.

以下を入力&実行し、結果を確認せよ。(例 4.4.8.)

```
#include<stdio.h>
int main(void){
    int i,j;
    printf("      | 1| 2| 3| 4| 5| 6| 7| 8| 9|¥n");
    printf("-----+---+---+---+---+---+---+---+---+---+¥n");
    for(i=1; i<=9; i++){
        printf("%d の段|",i);
        for(j=1;j<=9; j++)
            printf("%2d|",i*j);
        printf("¥n");
    }
    return 0;
}
```

## 実行結果

	1	2	3	4	5	6	7	8	9
1 の段	1	2	3	4	5	6	7	8	9
2 の段	2	4	6	8	10	12	14	16	18
3 の段	3	6	9	12	15	18	21	24	27
4 の段	4	8	12	16	20	24	28	32	36
5 の段	5	10	15	20	25	30	35	40	45
6 の段	6	12	18	24	30	36	42	48	54
7 の段	7	14	21	28	35	42	49	56	63
8 の段	8	16	24	32	40	48	56	64	72
9 の段	9	18	27	36	45	54	63	72	81

# for 文

この例の場合、1つめの for 文 (6 から 11 行) では変数として  $i$  を使い、2つめの for 文 (8,9 行) の変数として  $j$  を使っている。

もし、同じ変数  $i$  のみで行うと

実行結果

```
      | 1| 2| 3| 4| 5| 6| 7| 8| 9|
-----+---+---+---+---+---+---+---+---+---+
1の段| 1| 4| 9|16|25|36|49|64|81|
```

となって終わる。

# for 文の例

1~ $n$  までの和を求めるプログラムを考える。

プログラム : ( $n = 4$  のとき、for 文を用いた例)

```
#include<stdio.h>
int main(void){
    int sum=0,i,n=4;
    for(i=1;i<=n;i++){
        sum+=i;
    }
    printf("1 から%d までの和は%d です。¥n",n,sum);
    return 0;
}
```

# 分岐文

分岐文には `break` 文 と `continue` 文がある。

詳しくは次回紹介するが、ここでは `break` 文の例で少し紹介する。

# 分岐文

課題 J.1 を以下のように変える。

```
#include<stdio.h>
int main(void){
    int i,a;
    printf("自然数を入力してください:");
    scanf("%d",&a);
    for(i=0;i<a;i++){
        if(i==5){
            break;
        }
        printf("%d 回目のループ ¥n",i+1);
    }
    return 0;
}
```

# 本日の課題

- ・課題 J.1.
- ・課題 J.2.
- ・課題 J.3.
- ・課題 J.4. (プリント 71 ページ 課題 4.8.11.)

提出方法はメールで [yoshiyuki-mori@ous.ac.jp](mailto:yoshiyuki-mori@ous.ac.jp) まで送る。

**注意！6月からアドレス変更。**

件名は学生番号と課題 J. と書き、

例) S24M999 課題 J.

ファイルを添付し送ること。

締め切りは7月6日 23:59。