



知能情報工学演習I 第10回 (C言語第4回)

岩村雅一

masa@cs.osakafu-u.ac.jp

C言語の予定

7. 5月31日 プログラミング環境(テキスト1,2章)
8. 6月7日 変数とデータ型(3章)、演算子(4章)
9. 6月14日 コンソール入出力(6章)、配列(3章)、
数学処理の標準ライブラリ(11章)
10. 6月21日 制御文1(テキスト5章)
11. 6月28日 制御文2(テキスト5章)
12. 7月5日 関数1(テキスト7章)、
プリプロセッサ(テキスト10章)
13. 7月12日 応用プログラム

本日のメニュー

■ 制御文

□ 条件分岐

■ if文

■ switch文

□ 繰り返し

■ for文

■ while文

■ do-while文

□ ループの終了

■ break

■ continue

if文

- 条件を満たすときに命令を実行する

```
int a;  
a=10;  
if (a==10)
```

```
printf("aは10です。¥n ");
```

命令

- 条件の書き方(関係・等価演算子、テキストP.78)

- == 等しい

- != 等しくない

- > 大なり

- < 小なり

- >= 等号を含む大なり(\geq)

- <= 等号を含む小なり(\leq)

if-else文

- 条件を満たすと命令Aを実行し、満たさないときは命令Bを実行する

```
int a;  
scanf("%d", &a);  
if (a==10)
```

```
    printf("aは10です。¥n ");
```

```
else
```

```
    printf("aは10ではありません。¥n ");
```

命令A

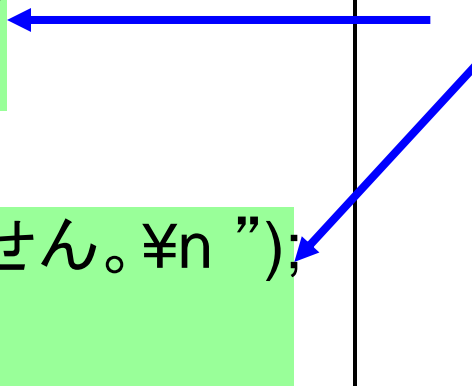
命令B

複合文(ブロック)

- 2行以上の文を実行したいとき

```
int a;  
scanf("%d", &a);  
if (a==10) {  
    printf("aは10です。¥n ");  
    printf("へへーん。¥n ");  
} else {  
    printf("aは10ではありません。¥n ");  
    printf("ほほーん。¥n ");  
}
```

ブロック



if文の入れ子

■ if文の中にif文

```
int a;  
scanf("%d", &a);  
if (a==10)  
    printf("aは10です。¥n ");  
else {  
    if (a==11)  
        printf("aは11です。¥n ");  
    else  
        printf("aは10でも11でもありません。¥n ");  
}
```

if文のサンプルプログラム

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int a;
```

```
    scanf("%d", &a);
```

```
    if (a==10) {
```

```
        printf("Bingo!!\n");
```

```
    } else {
```

```
        if (a<10) {
```

```
            printf("もっと大きい。 \n");
```

```
        } else {
```

```
            printf("もっと小さい。 \n");
```

```
        }
```

```
    }
```

```
    return 0;
```

```
}
```


switch文

```
int a;  
scanf("%d", &a);  
  
switch(a) {  
case 10:  
    printf("aは10です。¥n");  
    break;  
case 11:  
    printf("aは11です。¥n");  
    break;  
default:  
    printf("aは10でも11でもありません。¥n");  
}
```

for文

■ 条件を満たす間繰り返す

```
#include <stdio.h>
```

```
int main(void) {
```

```
int i;
```

```
for (i=0; i<3; i++) {
```

```
    printf("i=%d\n", i);
```

```
}
```

```
return 0;
```

```
}
```

初期化式

条件式

再設定式(増分)

for文

実行
i=0

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

出力

i=0

i=1

i=2

for文

i++を実行

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

i=1
↑
i=0

出力

```
i=0  
i=1  
i=2
```

for文

条件のチェック → 条件を満たす

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

i=1

出力

```
i=0  
i=1  
i=2
```

for文

実行
i=1

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

出力

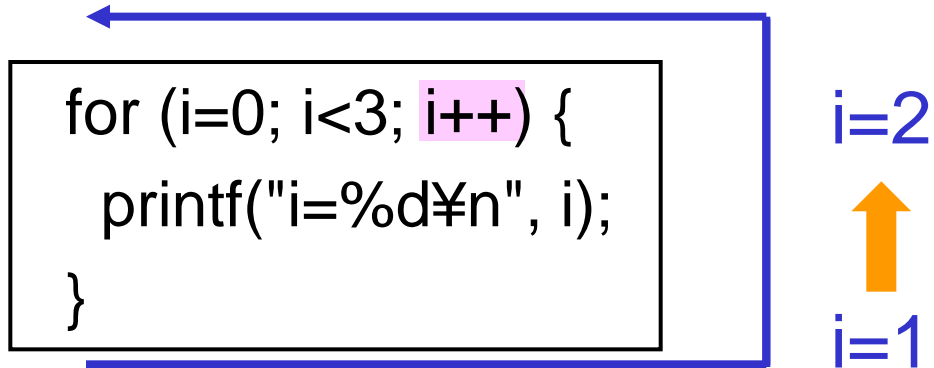
i=0

i=1

i=2

for文

i++を実行



出力

```
i=0  
i=1  
i=2
```

for文

条件のチェック → 条件を満たす

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

i=2

出力

```
i=0  
i=1  
i=2
```


for文

実行
i=2

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

出力

i=0

i=1

i=2

for文

i++を実行

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

i=3



i=2

出力

i=0

i=1

i=2

for文

条件のチェック →

条件を満たさないので
for文のループを終了

i=3

次の
処理へ

```
for (i=0; i<3; i++) {  
    printf("i=%d¥n", i);  
}
```

出力

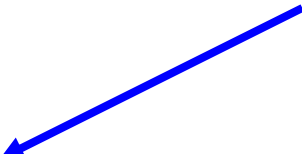
```
i=0  
i=1  
i=2
```

while文

- 条件を満たす間繰り返す
 - 条件式のみ
 - 下の例はさっきのfor文と等価

```
int i;  
i=0;  
while (i<3) {  
    printf("i=%d¥n", i);  
    i++;  
}
```

条件式



do-while文

- 条件を満たす間繰り返す
 - 基本的にwhile文と同じ
 - ただし、条件判定は後なので、必ず1回は実行する
 - 下の例はさっきのfor文、while文と等価

```
int i;  
i=0;  
do {  
    printf("i=%d¥n", i);  
    i++;  
} while(i<3);
```

条件式

セミコロンが必要

ループを抜け出す方法

■ break

- switch文で出てきた
- 一番内側のループを抜ける

■ continue

- 処理を1回だけスキップする

ループを抜け出す方法のサンプルプログラム

```
#include<stdio.h>
```

```
int main(void){
```

```
    int i;
```

```
    printf("break文の場合¥n");
```

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

```
        if (i==5) {
```

```
            break;
```

```
        }
```

```
        printf("i = %d¥n", i);
```

```
    }
```

```
    printf("continue文の場合¥n");
```

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

```
        if (i==5) {
```

```
            continue;
```

```
        }
```

```
        printf("i = %d¥n", i);
```

```
    }
```

```
    return 0;
```

```
}
```

サンプルプログラムの出力結果

■ break文の場合

i = 0
i = 1
i = 2
i = 3
i = 4

i = 5以降
表示されない

■ continue文の場合

i = 0
i = 1
i = 2
i = 3
i = 4
i = 6
i = 7
i = 8
i = 9

i = 5だけ
表示されない

