

課題に関する注意事項

情報処理演習 (4)条件分岐と繰り返しの 組み合わせ

知能システム学 准教授
万 偉偉(ワン ウエイウェイ)

- ・ プログラムの正しさ
- ・ 作成履歴・感想を細かく書くこと
- ・ 複数の実行の結果で検証すること
- ・ スクリーンショットを付けること
- ・ インデント

以上の内容をうまく対応できなければ再提出対象となる可能性が高い！

今日の内容

- ・ 繰り返し文と条件分岐文の組み合わせ
- ・ break文
- ・ continue文
- ・ switch-case文

繰り返しと分岐

- ・ 繰り返しや分岐は自由に「**入れ子**」に出来る

```
int i, num = 120;

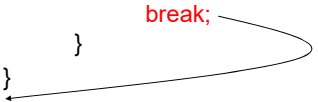
for (i = 1; i < num; i++) {
    if (num % i == 0) {
        printf("%d は %d で割り切れる. ¥n", num, i);
    }
}
```

約数を全て求める

繰り返しを中断するには

- ・ 繰り返し処理の途中で不都合が起こった
- ・ 繰り返しの残りを処理する必要がなくなったなど

```
int i, num = 120;
for (i = num-1; i >= 1; i--) {
    if (num % i == 0) {
        printf("%d は %d で割り切れる. ¥n", num, i);
        break;
    }
}
```



break の使い方

- ・ ループ(繰り返し)を即座に抜ける
 - 使えるのは for, while, do-while
- ・ ループを入れ子にしていたときは、最も**内側のループだけ**を抜ける
 - 二重ループを抜けるには**テクニックが必要**

```
for (i = 0; i < 10; i++) {
    for (j = 0; j < 10; j++) {
        if (何かの条件) {
            break;
        }
    }
    /* break ではここへ抜けてくる */
}
```

条件文とは

- ・ <, >, ==, != などの等号・不等号も計算式である
 - 条件が満たされれば 1, 満たされなければ 0
 - 以下のような計算もOK

```
int a, b, c;
```

```
a = (b == c);
if (a) {
    /* if (b == c) と同じ
    /* 何かの処理 */
}
```

無限ループ

- ・ while(**1**) {
.....
}
- ・ for (**;;**) {
.....
}
- ・ ループを抜ける手段(break)がないと使えない(プログラムは永遠に走り続ける)

次の処理へ進める

- ループ内の続きの処理を省略し、次の処理へ進める場合 `continue` を使う

```
for(i=0; i<10; i++){
    printf("何か整数値を入力して下さい。¥n");
    scanf("%d",&n);
    if( n < 0){
        continue;
    }
    /* n が負でないときにのみ行う処理 */
}
```

i に1を足す動作は行われる。

数値に合わせて分岐する

- 数値が 1 のとき, 2 のとき, 3 のときの3つの場合について異なる処理をさせたい

```
switch (a) {
    case 1:
        /* 1 のときの処理 */
        break;
    case 2:
        /* 2 のときの処理 */
        break;
    case 3:
        /* 3 のときの処理 */
        break;
}
```

「その他」の扱い

```
switch (a) {
    case 1:
        /* 1 のときの処理 */
        break;
    case 2:
        /* 2 のときの処理 */
        break;
    default:
        /* 1 でも 2 でもないときの処理 */
        break;
}
```

- 条件に合わないものは `default` で拾うことができる

switch-case と if

```
if (a == 1) {
    /* 1 のときの処理 */
}
else if (a == 2) {
    /* 2 のときの処理 */
}
else {
    /* 1 でも 2 でもないときの処理 */
}
```

- if 文でもほぼ同じことが出来る

switch-case の注意

```
switch (a) {
    case 1:
        /* 1 のときの処理 */
        break;
    case 2:
        /* 上に break がないと, 1 か 2 のときの処理 */
        break;
    default:
        /* 1 でも 2 でもないときの処理 */
        break;
}
```

- `break` を忘れると, case があってもそのまま下へ落ちる

switch-case の使い方

```
switch (a) {
    case 1:
    case 2:
        /* 1 か 2 のときの処理 */
        break;
    case 3:
    case 4:
        /* 3か 4 のときの処理 */
        break;
    default:
        /* 1~4 でないときの処理 */
        break;
}
```

- 「そのまま落ちる性質」を使う

今日学んだ内容

- 繰り返し文と条件分岐文の組み合わせ
- `break` 文
- `continue` 文
- 条件文の返り値 例: `1==2` の返り値は?
- 無限ループ `while(1)`, `for(;;)`
- `switch-case` 文