

第4章

繰り返し文と繰り返し制御文

目次

- while文
- do-while文
- for文と拡張for文
- 制御文のネスト
- 繰り返し制御文

●while文

構文

```
while(条件式){  
    処理分; //条件がtrueの場合に処理分が実行される  
}
```

ポイント

繰り返す処理回数が未確定な時に使用する

ループと無限ループ

- ・無限ループとは

`num++;`がなければ初期値0のままなのでずっと繰り返しが続いてしまう。

- ・減算でループ

3行目を`int num=5;`と記述することでループ可能。

do-while文

構文

```
do{  
  処理文;  
}while(条件式);
```

ポイント

while文と同様指定された条件が成立する間繰り返す。
一度繰り返し処理を行ってから条件判定が行われる。

●for文と拡張for文

構文

```
for(式1;式2;式3;){  
  処理文;  
}
```

ポイント

処理回数が確定している時に使用する。

拡張for文

構文

```
for(変数宣言:参照変数名){  
  処理分;  
}
```

ポイント

参照変数名で指定した配列などがもつ要素の数だけ処理が行われる。

→カウンタ変数の宣言なし

●制御文のネスト

ネスト・・・入れ子

ポイント

while文の中にfor文を記述できる。

→入れ子にしすぎると処理の流れが複雑になるため、注意

●繰り返し制御文

ある特定の条件が発生した場合に繰り返しから抜けるようにするための文のこと

- break文
- continue文

break文、continue文

- break文

現在実行中の繰り返し処理を中断して抜け出す時に使用する。

- continue文

現在実行中のブロック内の残りの処理をスキップして条件式制御を移し、さらに繰り返し処理を続ける時に使用する。

ラベル

入れ子になっているfor文の内側からはbreak文やcontinue文では抜け出せない。

→break ラベル名; と宣言することで抜け出せる！