

プログラミング言語基礎

[その10]

制御文 を使ってみよう!

- 「繰り返し」編 for文, do while文, while文
- 「条件分岐」編 if文, switch文

http://www.ipc.fukushima-u.ac.jp/~p058
 福島県立医科大学 物理 吉田 宏
 E-Mail: yoshidah@fmu.ac.jp

2004/6/22

プログラミング言語基礎(その10)

1

制御文

繰り返しと条件分岐

「繰り返し」編

歩く



足を動かす動作の繰り返し

疲れるまで歩こう! (とりあえず歩く) **do while文**
 10000歩 歩こう! (必ず歩く) **for文**
 疲れてなかったら, 歩き続けよう! **while文**
 (歩かないかも (^_^;))

2004/6/22

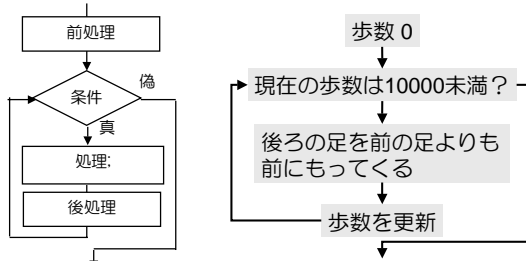
プログラミング言語基礎(その10)

2

for文

```
for (前処理; 条件; 後処理) {
    処理;
}
```

10000歩 歩く



2004/6/22

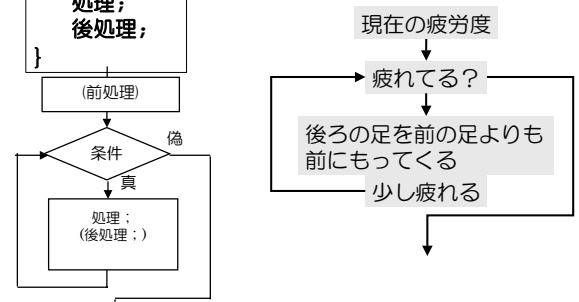
プログラミング言語基礎(その10)

3

while文

```
(前処理;)
while (条件) {
    処理;
    後処理;
}
```

疲れてなかったら 歩く

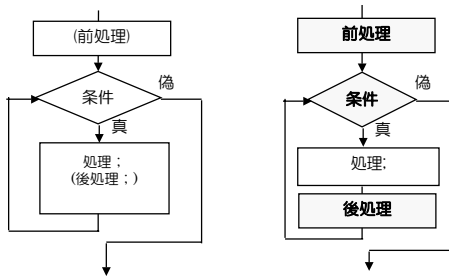


2004/6/22

プログラミング言語基礎(その10)

4

while文とfor文



2004/6/22

プログラミング言語基礎(その10)

5

while文 ex0209.c

```
1:/* ex0209.c 角度を「度」から「ラジアン」に変換するプログラム */
2:/* 作成者 xxxx */
3:/* 作成日 2004/06/23 */
4:
5:#include <stdio.h> /* stdio.hをインクルードする */
6:int main() /* main関数 */
7:{
8:  int degree=0; /* 整数型 degreeの宣言 */
9:  float PI=3.141592654; /* PI(円周率), radianの宣言 */
10:
11:  while (degree<=180) {
12:      radian = degree * PI / 180; /* ラジアンの変換 */
13:      printf("%4d %8.3f\n", degree, radian); /* 結果の出力 */
14:      degree+=10; /* degreeを10増やす */
15:  }
16:  return 0; /* 終了 */
17:}
```

2004/6/22

プログラミング言語基礎(その10)

6

while文からfor文への書き換え

```
1:/* ex0401.c 角度を「度」から「ラジアン」に変換するプログラム */
2:/* 作成者 */
3:/* 作成日 */
4:
5:#include <stdio.h> /* stdio.hをインクルードする */
6:int main() /* main関数 */
7:{
8:  int degree; /* 整数型 degreeの宣言 */
9:  float radian; /* 単精度実数型 radianの宣言 */
10:  float PI=3.141592654; /* PI 円周率 */
11:
12:  for (degree=0; degree<=180; degree+=10) {
13:      radian = degree*PI/180; /* ラジアンの変換 */
14:      printf("%4d %8.3f\n", degree, radian); /* 結果の出力 */
15:  }
16:  return 0; /* 終了 */
17:}
```

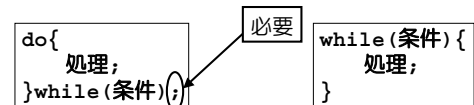
Try 課題4.3&4.4

2004/6/22

プログラミング言語基礎(その10)

7

do while文



while文とdo while文の違い

while 文 慎重派	条件文が始めにくる (前判定) 次の「処理」を繰り返し行うか? 1度も実行されないこともある
do while 文 行動派	条件文が後にくる (後判定) 今の行った「処理」をもう1度繰り返す? 少なくとも1度は実行される

2004/6/22

プログラミング言語基礎(その10)

8

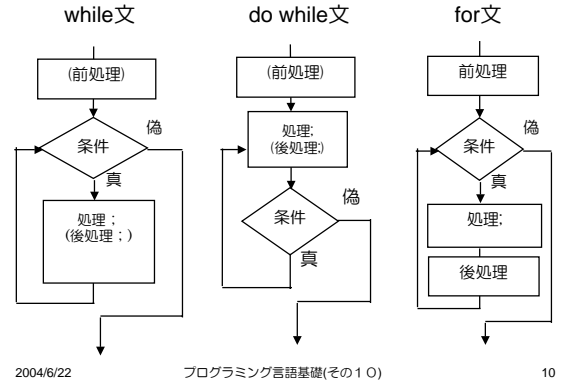
while文からdo while文への書き換え

```

1:/* ex0404.c 角度を「度」から「ラジアン」に変換するプログラム */
2:/* 作成者 xxxxx */
3:/* 作成日 2004/06/23 */
4:
5:#include <stdio.h> /* stdio.hをインクリードする */
6:int main() /* main関数 */
7:{
8:  int degree=0; /* 整数型 degreeの宣言 */
9:  float PI=3.141592654, radian; /* PIの宣言, radian */
10:
11:  do{ /* とりあえずやっちゃえ! */
12:    radian = degree*PI/180; /* ラジアンの変換 */
13:    printf("%4d %8.3f\n", degree, radian); /* 結果の出力 */
14:    degree+=10; /* degreeを10増やす */
15:  }while(degree<=180); /* 条件判定 */
16:  return 0; /* 終了 */
17;}
    
```

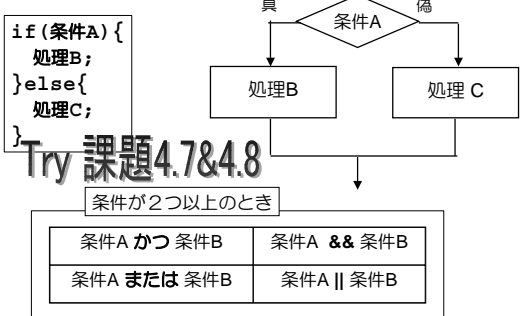
Try 課題4.5&4.6

繰り返しの制御文



「条件分岐」編

if文



Try 課題4.7&4.8

複数の条件分岐

成績判定

aが70以上 ⇒ h='A' (^_^)Y
aが60以上70未満 ⇒ h='B' (^_^)
aが50以上60未満 ⇒ h='C' (':;)
aが50未満 ⇒ h='D' (':;)

```

int a;
char h;
.....
if (a>=70){
  h='A';
} else if (a>=60){
  h='B';
} else if (a>=50){
  h='C';
} else {
  h='D';
}
    
```

Try 課題4.9

if文の条件式

```

if(条件式){
  処理1;
}else{
  処理2;
}
    
```

条件式

- 通常の場合式
 - <, >, <=, >=, ==, !=
- 整数値
 - 0 のとき偽
 - 0以外 のとき真

```

/* 2のときの例 */
int a;

if (a){
  /* aがゼロでなければ */
  処理1; /* 真 */
}else{
  /* aがゼロならば */
  処理2; /* 偽 */
}
    
```

πを計算しよう!

ζ関数

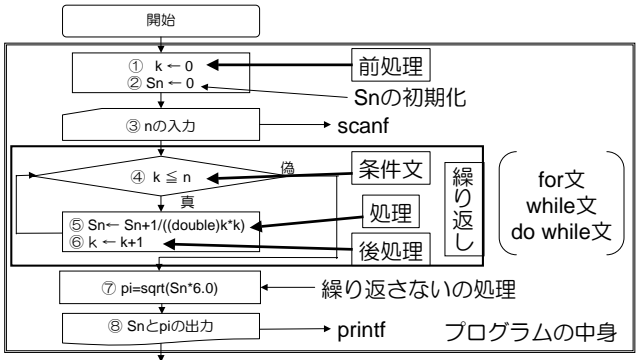
$$\zeta(2) = \sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6} \approx \sum_{k=1}^n \frac{1}{k^2} = S_n$$

$$\zeta(4) = \sum_{k=1}^{\infty} \frac{1}{k^4} = \frac{\pi^4}{90} \approx \sum_{k=1}^n \frac{1}{k^4} = S_n$$

参考 ex0301.c

$$S_n = 1 + 2 + 3 + 4 + \dots + n = \sum_{k=1}^n k$$

フローチャート



プログラムの作成

```

/* πの計算 ex0411.c */
/* 作成者 xxxxxx */
/* 作成日 2004/06/23 */
#include <stdio.h>
#include <math.h>

int main()
{
  各変数の宣言;

  プログラムの中身

  return 0;
}
    
```

Try 課題4.11