

プログラミング言語基礎

[その8]

今日の目標！

<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

2004/6/8

プログラミング言語基礎(その8)

1

2004/6/8

プログラミング言語基礎(その8)

2

1. 配列ってなんだろう？
 - 配列を使ったプログラムを作ろう！
2. フローチャートを書ける, 読めるようになろう！
3. 制御文 を使ってみよう！
 - for文, do while文, while文

1. 配列

同じ変数型の集まり ベクトルや行列など

44人の「プログラミング言語基礎」の期末テストの平均値
配列を使わないと

```
int a01,a02,a03,a04,a05,a06,a07,a08,a09,a10,a11;
int a12,a13,a14,a15,a16,a17,a18,a19,a20,a21,a22;
int a23,a24,a25,a26,a27,a28,a29,a30,a31,a32,a33;
int a34,a35,a36,a37,a38,a39,a40,a41,a42,a43,a44;
int heikin,total;
```

```
total=a01+a02+a03+a04+a05+a06+a07+a08+a09+a10+ a11
+a12+a13+a14+a15+a16+a17+a18+a19+a20+a21+a22
+a23+a24+a25+a26+a27+a28+a29+a30+a31+a32+a33
+a34+a35+a36+a37+a38+a39+a40+a41+a42+a43+a44;
heikin=total/44.0;
```

2004/6/8

プログラミング言語基礎(その8)

3

配列を使うと

```
/* 変数を宣言するとき */
int heikin, total, i, a[44];

/* 合計&平均を求める */
i, total の最初の値を設定する
i=total=0; (初期化)
while(i<44){ iが44になったら, 演算を止める
total+=a[i]; totalにa[i]のデータを加算する
i++; iを1増やす
}
heikin=total/44.0; total/440.0;
```

44人->440人とするには...

2004/6/8

プログラミング言語基礎(その8)

4

配列の宣言の仕方

```
int a[10]; /* 整数型の配列aとして要素数10の変数を宣言 */
float a; /* 同じ関数内で同名の変数は宣言できない */
```

この宣言で
a[0], a[1], a[2], a[3], a[4], a[5], a[6], a[7], a[8], a[9]
の整数型変数が使用できる。

- 配列の大きさ N (=10) を宣言文で指定する
- 0番目からN-1番目まで(a[0],...,a[9]) が変数として使える
- 配列として宣言した変数は, 同じ関数内で別の変数の名前と重複してはいけない

Try 課題 2.10&11

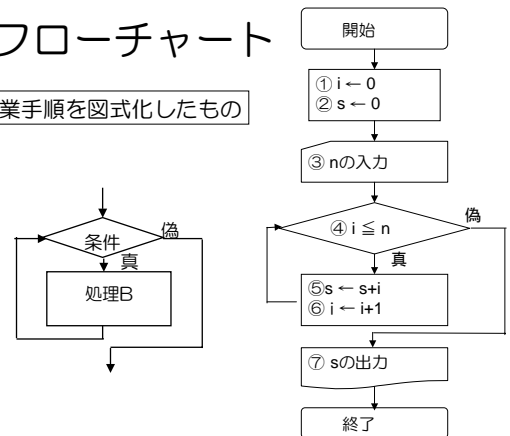
2004/6/8

プログラミング言語基礎(その8)

5

2. フローチャート

作業手順を図式化したもの

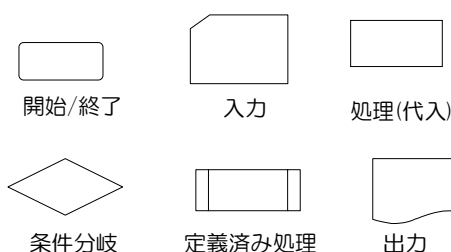


2004/6/8

プログラミング言語基礎(その8)

6

フローチャートの部品

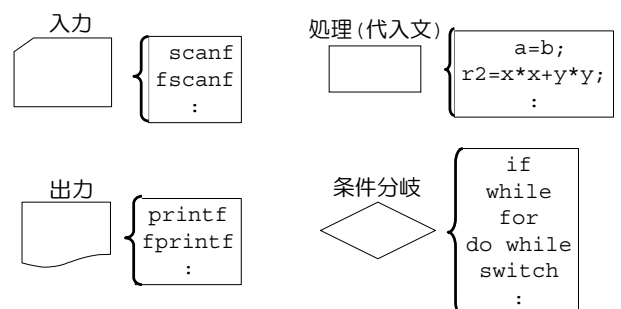


2004/6/8

プログラミング言語基礎(その8)

7

部品との対応



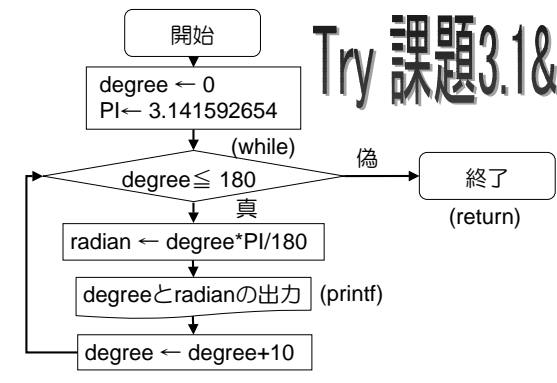
2004/6/8

プログラミング言語基礎(その8)

8

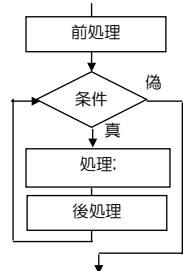
ex0208.cのフローチャート

Try 課題3.1&3.2



for文

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



for文

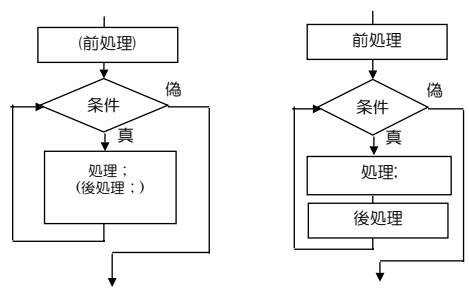
```
/* ex0400.c for文の例 */
/* 作成者 xxxx */
/* 作成日 2004/06/09 */

#include <stdio.h>
int main(){
    int i,j;

    for(i=0;i<10;i++){
        for(j=0;j<=i;j++) printf("*");
        printf("\n");
    }

    for(i=9;i>=0;i--){
        for(j=0;j<=i;j++) printf("*");
        printf("\n");
    }
    return 0;
}
```

while文とfor文



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 xxxx */
3:/* 作成日 2004/06/09 */
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:}
```

do while文



while文とdo while文の違い

- while 文** 条件文が始めにくる (前判定)
次の繰り返しを行うか?
1度も実行されないこともある
- do while 文** 条件文が後にくる (後判定)
今の繰り返しを行うか?
少なくとも1度は実行される

繰り返しの制御文

