

## プログラミング基礎演習A



文字型変数  
定数  
書式指定

## 文字コード

半角  
スペース

文字を表す番号

0: NUL	16	32: SP	48: 0	64: @	80: P	96: `	112: p
1	17	33: !	49: 1	65: A	81: Q	97: a	113: q
2	18	34: "	50: 2	66: B	82: R	98: b	114: r
3	19	35: #	51: 3	67: C	83: S	99: c	115: s
4	20	36: \$	52: 4	68: D	84: T	100: d	116: t
5	21	37: %	53: 5	69: E	85: U	101: e	117: u
6	22	38: &	54: 6	70: F	86: V	102: f	118: v
7	23	39: '	55: 7	71: G	87: W	103: g	119: w
8: BS	24	40: (	56: 8	72: H	88: X	104: h	120: x
9: HT	25	41: )	57: 9	73: I	89: Y	105: i	121: y
10: LF	26	42: *	58: :	74: J	90: Z	106: j	122: z
11	27	43: +	59: ;	75: K	91: [	107: k	123: {
12	28	44: ,	60: <	76: L	92: ¥	108: l	124:
13: CR	29	45: -	61: :	77: M	93: ]	109: m	125: }
14	30	46: .	62: >	78: N	94: ^	110: n	126: ~
15	31	47: /	63: ?	79: O	95: _	111: o	127: DEL

すべて半角文字 青字は制御文字<sub>2</sub>

## 文字型変数

char

◆ 半角文字(文字コード)が入る変数の型

- ◆ 変数 = 文字コード (整数)
- ◆ 変数 = '半角文字'

全角文字は  
char型では  
扱えない

76の代わりに'L',  
'A'の代わりに65  
を代入しても  
結果は同じ

```
int main()
{
    char size, class;

    size = 76;
    class = 'A';
}
```

76 'A'  
size class

## printf関数 / scanf関数における文字型変数の書式指定

◆ 変換指定子は%c

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

```
int main()
{
    char letter;

    printf("文字を入力\n");
    scanf("%c", &letter);
    printf("letterは%cです\n", letter);
    printf("letterの文字コードは%dです\n", letter);
    if(letter == 65) {
        printf("letter is A\n");
    }
}
```

文字を入力  
A  
letterはAです  
letterの文字コードは65です  
letter is A

'A'  
letter

'A'でもよい

## プログラム例(1)

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

```
int main()
{
    char letter;

    letter = 'T';
    printf("Tの文字コードは%dです\n", letter);
    letter = letter + 1;
    printf("Tの次の文字は%cです\n", letter);
}
```



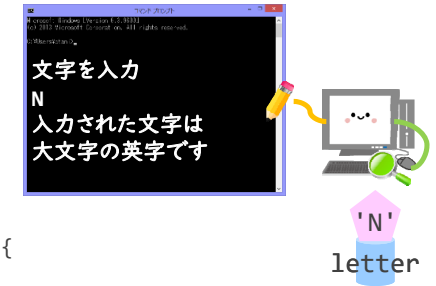
5

## プログラム例(2)

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

```
int main()
{
    char letter;

    printf("文字を入力\n");
    scanf("%c", &letter);
    printf("入力された文字は\n");
    if((letter >= 'A') && (letter <= 'Z')) {
        printf("大文字の英字です\n");
    } else {
        printf("大文字の英字ではありません\n");
    }
}
```



6

## オブジェクト形式マクロ

... **#define** マクロ名 文字列

- ◆ これ以降の「マクロ名」をすべて「文字列」に置換する
- ◆ 定数をオブジェクト形式マクロで定義すると便利
- ◆ マクロ名は大文字にするのが慣習



7

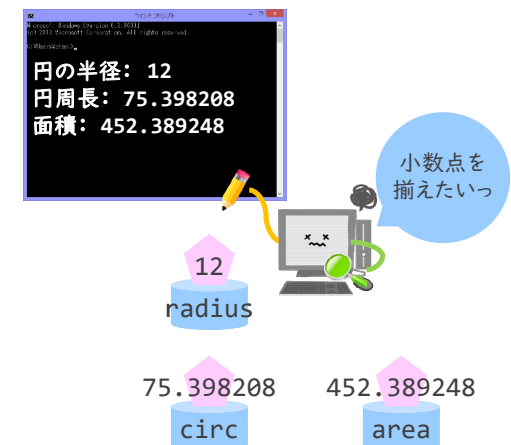
## プログラム例(3)

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

```
#define PI 3.141592
```

```
int main()
{
    double radius, circ, area;

    printf("円の半径: ");
    scanf("%lf", &radius);
    circ = 2 * radius * PI;
    area = radius * radius * PI;
    printf("円周長: %f\n", circ);
    printf("面積: %f\n", area);
}
```



8

## printfにおける書式指定

```
printf("フォーマット文字列", 変数, 変数, ...)
```

### ◆ 変換指定子

- ◆ **%d** : 整数(int型)を10進数で表示
- ◆ **%f** : 実数(float型,double型)を表示
- ◆ **%c** : 文字(char型)を表示
- ◆ **%s** : 文字列を表示
- ◆ **%X** : 符号なし整数(unsigned型)を16進数で表示

↑

┌ ├ └	➤ d, f, s, Xの前に <b>ピリオド</b> + 整数n	$\left\{ \begin{array}{l} \text{d, X : 最小でn桁を表示} \\ \text{f : 小数点以下をn桁で表示} \\ \text{s : 最大でn文字を表示} \end{array} \right.$
	➤ %の後に整数n : 最小フィールド幅をnとして表示	

9

## プログラム例(3)改

```
#include <stdio.h>

#define PI 3.141592

int main()
{
    double radius, circ, area;

    printf("円の半径: ");
    scanf("%lf", &radius);
    circ = 2 * radius * PI;
    area = radius * radius * PI;
    printf("円周長: %6.2f¥n", circ);
    printf("面積: %6.2f¥n", area);
}
```



10

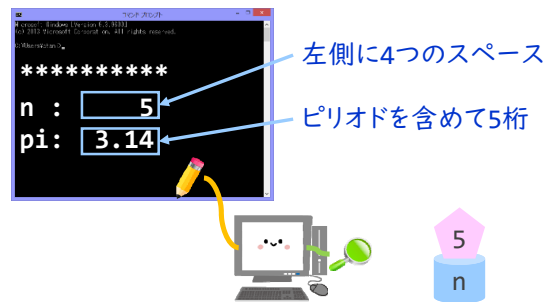
## プログラム例(4)

```
#include <stdio.h>

#define PI 3.141592

int main()
{
    int n = 5;

    printf("*****¥n");
    printf("n : %5d¥n", n);
    printf("pi: %5.2f¥n", PI);
}
```



11

## 演習

$\theta = -15^\circ \sim 15^\circ$ の範囲の $\sin(\theta)$ の値を1度刻みで右の実行例のように表示するプログラムを作成せよ。ただし、値は小数第3位まで表示し、縦方向に小数点が揃うようにすること。また、円周率の値はオブジェクト形式マクロで定義すること。プログラム名はe9とすること。

```
sin(-15) : -0.259
sin(-14) : -0.242
sin(-13) : -0.225
sin(-12) : -0.208
sin(-11) : -0.191
sin(-10) : -0.174
sin(-9) : -0.156
sin(-8) : -0.139
sin(-7) : -0.122
sin(-6) : -0.105
sin(-5) : -0.087
sin(-4) : -0.070
sin(-3) : -0.052
sin(-2) : -0.035
sin(-1) : -0.017
sin(0) : 0.000
sin(1) : 0.017
sin(2) : 0.035
sin(3) : 0.052
sin(4) : 0.070
sin(5) : 0.087
sin(6) : 0.105
sin(7) : 0.122
sin(8) : 0.139
sin(9) : 0.156
sin(10) : 0.174
```

12

## 演習 ～手順～

1. 中身が空のメイン関数を書く
2. 必要なヘッダファイルは？
  - 入力や出力には**stdio.h**が必要
  - 数学関数の使用には**math.h**が必要
  - **#include**の行を書く
3. 必要な変数は？
  - 角度を格納する変数
  - 正弦値を格納する変数
  - 型と変数名を決めて変数を宣言する

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

int main()
{
    int theta;
    double sine;
}
```



変数名での  
スペルミスは  
かっこわるいぞ!

13

## 演習 ～手順～

4. オブジェクト形式マクロの定義は？
  - 円周率を定義する
5. 処理の順番は？
  - 以下をtheta=-15～15で繰り返す (**for**文始まり)
  - sin関数で正弦値を求め, sineに代入する
  - sineの値を表示する



ラジアンに  
変換したthetaを  
sin関数の  
引数にしよう

$\pi \text{rad} = 180^\circ$

14