# 精密工学科プログラミング基礎Ⅱ

## 第4回資料

## 今回の授業で習得してほしいこと:

- 文字列の扱い
  - 1次元配列の復習
  - 文字とアスキーコードの関係
- ファイル入出力の方法
- コマンドライン引数の使い方

資料のURL : http://www.den.t.u-tokyo.ac.jp/prog/

# 文字列とは？

- char 型の1次元配列 (文字の配列)
  - 文字列の最後には終りの記号 **'¥0'** が入っている

文字列 : Hi !

```
char st[5];
st[0] = 'H';
st[1] = 'i';
st[2] = ' ';
st[2] = '!';
st[3] = '¥0';
```

文字列にだけ使える
初期化の方法

```
char st[5] = "Hi !";
```

※ 一文字なら ' '，まとめてなら " " で囲む.

# 文字列の入出力

- 一文字ずつなら: <u>%c</u>
- まとめてなら: <u>%s</u> (スペース・改行で区切られる)
- 数字(アスキーコード)としてなら: %d

ある単語をスキャンしてプリント

```
char st[255];

scanf(”%s”, st);

printf(”%s¥n”, st);
```

一文字づつなら →

```
int i;
for(i=0; st[i] != ’¥0’; i++){
  printf(”%c”, st[i]);
}
printf(”¥n”);
```

# コマンドライン引数

- main 関数へ渡す文字列の引数
  - 例1 : emacs  test.c
    (emacs の main 関数へ開きたいファイル名を渡した)
  - 例2 : cp  file1  file2
    (cp へコピー元ファイルとコピー先ファイルを渡した)

引数の数+1

引数の値 (文字列の配列)

```
int main( int argc,  char *argv[ ] ){

  printf(”%d”, argc);
  printf(”%s¥n”, argv[1]);

}
```

※ 1番から

実行

```
% ./a.out  abc  efg
3
abc
```

0 番には
コマンド名が入っている

# 文字列から数値へ変換

- 文字列を数字に変換したいなら以下を使う
  - 整数 int 型 : atoi(文字列)
  - 小数 double 型 : atof(文字列)
  - #include <stdlib.h> を入れる

2つの文字列を整数に変換して足し算をする

```
char st1[ ] = ”123”;
char st2[ ] = ”100”;
int a,b;
a = atoi(st1);
b = atoi(st2);
printf(“%d¥n”, a+b);
```

# ファイル入出力

- ファイルからデータを読み書きする
  - 複数のファイルから入出力を行える.
    (リダイレクション “<”, ”>” は1つのファイル操作のみ)
  - オープンとクローズの操作が加わるだけで、
    使い方は printf, scanf と同じ

**ファイル test.txt から変数 a に値を読み込み**

```
int a;
FILE* f;
f = fopen(”test.txt”, ”r”);

fscanf(f, ”%d”, &a);

fclose(f);
```

(FILE* → ファイル型, ”r” → read)

**ファイル test.txt へ変数 a の値を書き出す**

```
int a = 10;
FILE* f;
f = fopen(”test.txt”, ”w”);

fprintf(f, ”%d”, a);

fclose(f);
```

(”w” → write)

# 課題 (1)

scanf を使って単語を読み込み以下を行え

1. 文字数のカウント
   – ”Word” なら 4 になる
2. 逆さ文字にして表示
   – ”Word” なら ”droW” になる
3. アスキーコードを表示
   – ’A’ と ’a’ ,  ’B’ と ’b’ の関係はどうなっている？
4. 大文字⇔小文字変換
   – ”Word” なら ”wORD” になる
5. 各アルファベットの出現を数える
   – 大文字・小文字の区別はなし
   – カウンタ26個は配列で用意する
   – ”Exercises” なら出力例は c:1, e:3, i:1, r:1, s:2, x:1

# 課題 (2)

以下のプログラムをコマンドライン引数を用いて作成せよ

1. 数列を引数として渡し、それらの和を計算する
   – ./a.out 1 2 3 4 とすると 10 が表示される
2. 整数列が書かれた2つのファイルを指定し,
   総和を計算せよ.
   – ./a.out  data1.txt  data2.txt とすると
     ファイル中の数の和が表示される.
3. 入力ファイルの文字列を,
   1文字おきに2つのファイルへ交互に出力する
   – ./a.out  in.txt  odd.txt  even.txt とすると奇数番目の文字は odd.txt へ，偶数番目は even.txt へ書き出される
4. 課題(2)-3で作成した2つのファイルを元に戻す
   – ./a.out  odd.txt  even.txt  out.txt とすると out.txt へ元のファイルが復元される