

# C言語講座 第4回

ファイルの入出力

A decorative graphic consisting of several horizontal lines of varying lengths and colors (teal, light blue, white) extending from the right side of the slide.

# ファイルとモード

- ファイルには「テキストファイル」と「バイナリファイル」の2種類があり、どちらのファイルをどのような形で開くのかを指定するのが、モードの役割である。
- テキストファイルはごく一般的なファイルであるが、バイナリファイルでは音声や画像を扱うことが可能である。

※今回はテキストファイルのみ紹介します。

# システム標準関数

- 第1回に習った”printf”や”scanf”は、もともと Microsoft Visual C++ 2010 Expressによって備え付けられた関数である。これを「システム標準関数」という。
- このシステム標準関数を利用することによって、キーボードからの入力や画面への出力に限らず、ファイルの入出力も可能になる。

# ファイル入出力のシステム関数

- fopen : ファイルを開く。
- fclose : ファイルを閉じる。
- fprintf : ファイルに書式付の文字列を書き込む。  
(書き込みモードに使用される)
- fscanf : ファイルから文字列を読み込み、指定された形式に変換する。(読み込みモードに使用される)
- fgets : ファイルから1行読みだす。
- fputs : ファイルに1行書き込む。

他にもいくつか存在する。モードの説明は後程。

# ファイルオープン・クローズの書き方

- ファイルオープン

```
FILE *変数名;
```

```
変数名 = fopen(“ファイル名”, “モード” );
```

- ファイルクローズ

```
fclose(変数名);
```

## 前スライドの説明

- FILE \***変数名**;  
FILE\*型ポインタ変数の宣言。
- **変数名** = fopen(“**ファイル名**”, “**モード**” );  
ファイルを開く操作の記述。
- fclose(**変数名**);  
開いたファイルを閉じる操作の記述。

変数名は自分で自由に決めて良い。

# モードの種類

- r : テキストファイルの読み込みモード。
- w : テキストファイルの書き込みモード。
- a : テキストファイルの追加モード。
- r+ : テキストファイルの更新モード。

# 例題1 : 4-1.c

- #include <stdio.h>

```
int main(){
```

```
FILE *fp;
```

```
char str[256];
```

```
fp = fopen("open.txt","w");
```

```
if(fp==NULL){
```

```
printf("ファイルオープン失敗。¥n");
```

```
}
```

```
fclose(fp);
```

```
return 0;
```

```
}
```

- ファイル「open.txt」はフォルダの中に作られているはず。
- 5行目のかっこ内を「"close.txt","r"」、「"close.txt","a"」にすると？

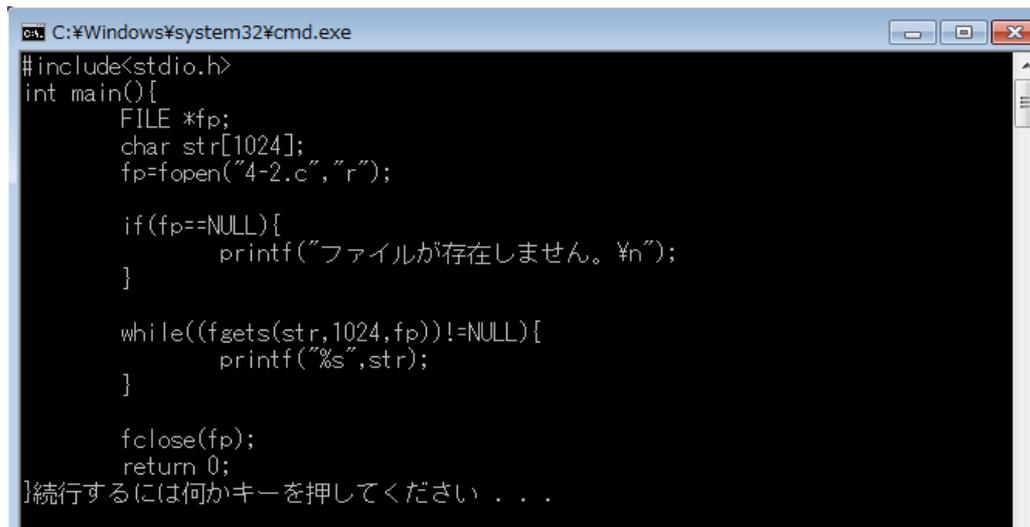
## 例題2 : 4-2.c

```
#include <stdio.h>
int main(){
FILE *fp;
char str[1024];
fp = fopen("4-2.c","r");

if(fp==NULL){
printf("ファイルが存在しません。¥n");
}

while((fgets(str,1024,fp))!=NULL){
printf("%s",str);
}

fclose(fp);
return 0;
}
```



```
cmd: C:\Windows\system32\cmd.exe
#include<stdio.h>
int main(){
FILE *fp;
char str[1024];
fp=fopen("4-2.c","r");

if(fp==NULL){
printf("ファイルが存在しません。¥n");
}

while((fgets(str,1024,fp))!=NULL){
printf("%s",str);
}

fclose(fp);
return 0;
}続行するには何かキーを押してください . . .
```

- 5行目の「4-2.c」を「4-1.c」や前回までの講座の例題のファイル名にすると？

# 例題1,2の補足

- 例題1

5行目のかっこ内を「" close.txt","r"」にしてデバッグすると、エラーが出力される。

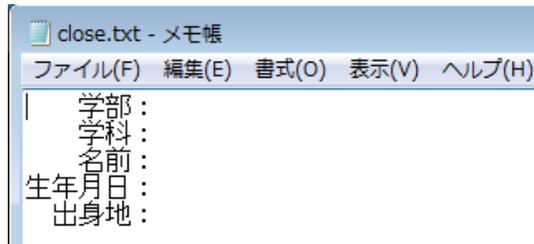
「" close.txt","a"」だと、フォルダに新しく close.txtが作られる。先に「" close.txt","a"」をデバッグし、「" close.txt","r"」でデバッグすると、何も起こらず終了する。

- 例題2

存在するファイル名であれば、そのファイルが読み込まれる。存在しないファイル名ならば、エラーで返される。

# 演習1：4-3.c

- 例題1で作った「open.txt」に、次のスライドのテンプレを使って自己紹介を書いてください。  
※書いたら保存させてください。
- その自己紹介をデバッグ画面及び「close.txt」に同時に表示させるプログラムを作成してください。



# 自己紹介のテンプレ

学部：

学科：

名前：

生年月日：

出身地：

## .txt以外にも . . .

- 先ほどは.txtファイル(メモ帳)を扱う入出力を紹介しましたが、実はワードやエクセルにも入出力させることができる。
- ここからは例として、エクセルに出力させるプログラムを学びたいと思います。

エクセルにファイルを作るには . . .

- 「ファイル名.csv」とするだけでOK。
- 試しに、メモ帳を起動し、適当に文章や数字を入力した状態で名前を付けて、.txtを.csvに変えて保存してみよう。

## 改行とセルの移動

- 改行はエクセルでもそのまま改行されて出力される。
- ただし、このままでは1個のセルにすべて入力されてしまう。隣のセルにも出力させるには、データ間を「,」で区切る必要がある。

## 例題3 : 4-4.c

```
#include<stdio.h>
int main(void){
    int x=12,y=15,z=19;
    FILE*fp;

    fp=fopen("4-4.csv","w");
    if(fp==NULL)
        printf("ファイルはありません\n");
    else{
        fprintf(fp,"x,%d\n",x);
        fprintf(fp,"y,%d\n",y);
        fprintf(fp,"z,%d\n",z);
        fclose(fp);}
    return 0;
}
```

## 例題3の補足

- 先ほどと同様に、プログラムによって出力させることができる。
- 「,」によって区切ることで隣のセルへの書き込みが可能になり、「¥n」によって改行させることができる。

## 最終問題：4-5.c

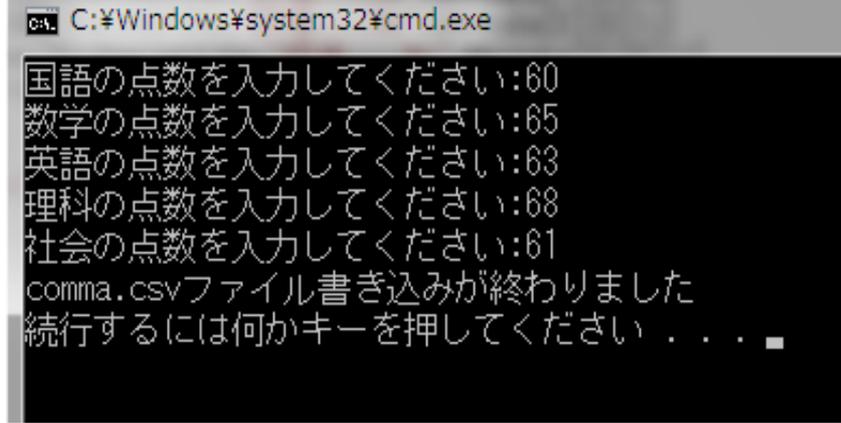
- 5科目の点数を入力し、それぞれの点数、合計、平均をエクセルに表として出力させ、平均点数によって何らかの評価をエクセルに表示させるプログラムを作成してください。

※このままでは難しいと思うので、次のページにソースコードの一部(笑)を載せてあるので、不足している部分を補いながら書いてください。

# ソースコード1

```
#include <stdio.h>

int main(void)
{
    FILE *fp;
    char *fname = "ファイル名と拡張子は?";
    char kamoku[5][7]={"国語","数学","英語","理科","社会"};
    char hyouka[5][16]={"神","優等生","凡人","マダオ","脳筋"};
    int sum=0;
    double ave=0;
    int i;
    int a[5];
    for(i=0;i<5;i++){
        printf("%sの点数を入力してください:",&kamoku[i][0]);
        scanf("%d",&a[i]);
        sum+=a[i];
    }
    ave=sum/5.0;
```



```
C:\Windows\system32\cmd.exe
国語の点数を入力してください:60
数学の点数を入力してください:65
英語の点数を入力してください:63
理科の点数を入力してください:68
社会の点数を入力してください:61
comma.csvファイル書き込みが終わりました
続行するには何かキーを押してください . . . .
```

## ソースコード2

```
fp = fopen( fname, "モードは何か?");
if( fp == NULL ){
printf( "%sファイルが開けません\n", fname );
return -1;
}
for(i=0;i<5;i++){
fprintf( fp, "%s,%d\n",&kamoku[i][0],a[i]);
}
fprintf( fp, "sum is,%d\n",sum);
fprintf( fp, "average is,%lf\n",ave);
if(ave>=90){fprintf(fp,"評価...,%s",&hyouka[0][0]);}
if(ave>=80&&ave<90){fprintf(fp,"評価...,%s",&hyouka[1][0]);}
if(ave>=70&&ave<80){fprintf(fp,"評価...,%s",&hyouka[2][0]);}
if(ave>=60&&ave<70){fprintf(fp,"評価...,%s",&hyouka[3][0]);}
if(ave<60){fprintf(fp,"評価...,%s",&hyouka[4][0]);}
```

	A	B	C
1	国語	60	
2	数学	65	
3	英語	63	
4	理科	68	
5	社会	61	
6	sum is	317	
7	average is	63.4	
8	評価...	マダオ	
9			

ここにファイルを閉じる処理を記述しよう

```
printf( "%sファイル書き込みが終わりました\n", fname );
return 0;
}
```

# 演習1解答例(1/2)

```
#include <stdio.h>
int main(){
FILE *fp,*gp;
char str[256];

fp = fopen("open.txt","r");
if(fp==NULL){
printf("ファイルオープンエラー");
}

gp = fopen("close.txt","w");
if(gp==NULL){
    printf("ファイルオープンエラー");
}
```

# 演習1解答例(2/2)

```
while(fgets(str,255,fp)){  
    printf("%s", str);  
    fputs(str,gp);  
}
```

```
printf("¥n");  
fclose(fp);  
fclose(gp);  
return 0;  
}
```

# 最終問題解答例(1/2)

```
#include <stdio.h>

int main(void)
{
    FILE *fp;
    char *fname = "4-5.csv";
    char kamoku[5][7]={"国語","数学","英語","理科","社会"};
    char hyouka[5][16]={"神","優等生","凡人","マダオ","脳筋"};
    int sum=0;
    double ave=0;
    int i;
    int a[5];
    for(i=0;i<5;i++){
        printf("%sの点数を入力してください:",&kamoku[i][0]);
        scanf("%d",&a[i]);
        sum+=a[i];
    }
    ave=sum/5.0;
```

# 最終問題解答例(2/2)

```
fp = fopen( fname, "w" );
if( fp == NULL ){
printf( "%sファイルが開けません\n", fname );
return -1;
}
for(i=0;i<5;i++){
fprintf( fp, "%s,%d\n",&kamoku[i][0],a[i]);
}
fprintf( fp, "sum is,%d\n",sum);
fprintf( fp, "average is,%lf\n",ave);
if(ave>=90){fprintf(fp,"評価...,%s",&hyouka[0][0]);}
if(ave>=80&&ave<90){fprintf(fp,"評価...,%s",&hyouka[1][0]);}
if(ave>=70&&ave<80){fprintf(fp,"評価...,%s",&hyouka[2][0]);}
if(ave>=60&&ave<70){fprintf(fp,"評価...,%s",&hyouka[3][0]);}
if(ave<60){fprintf(fp,"評価...,%s",&hyouka[4][0]);}

fclose( fp );

printf( "%sファイル書き込みが終わりました\n", fname );
return 0;
}
```