

C言語講座 第5回

ファイル入出力

担当:おさない、水谷、山崎

ファイル入出力とは？

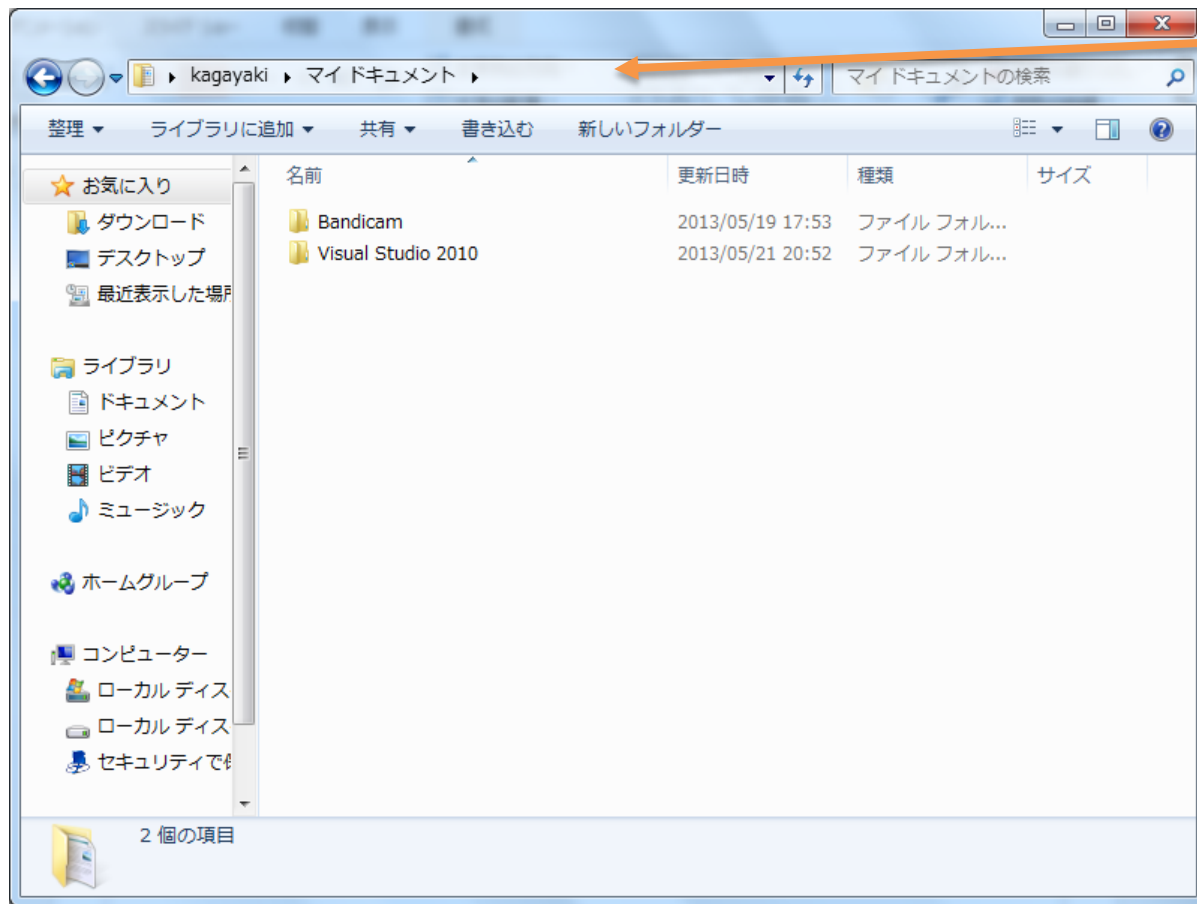
- ファイルから入力进行もらう
- ファイルに出力する
- 具体的には
 - 大きな配列を、統計データなどで初期化する時
 - プログラムの实行結果をグラフに表す时などに使われる

ファイル入出力のプログラム例1/5

- 何も説明していませんが、ファイル入出力がどんな動作をするのか実行してみましょう！
- まず、左下のスタートボタン→ドキュメント

ファイル入出力のプログラム例2/5

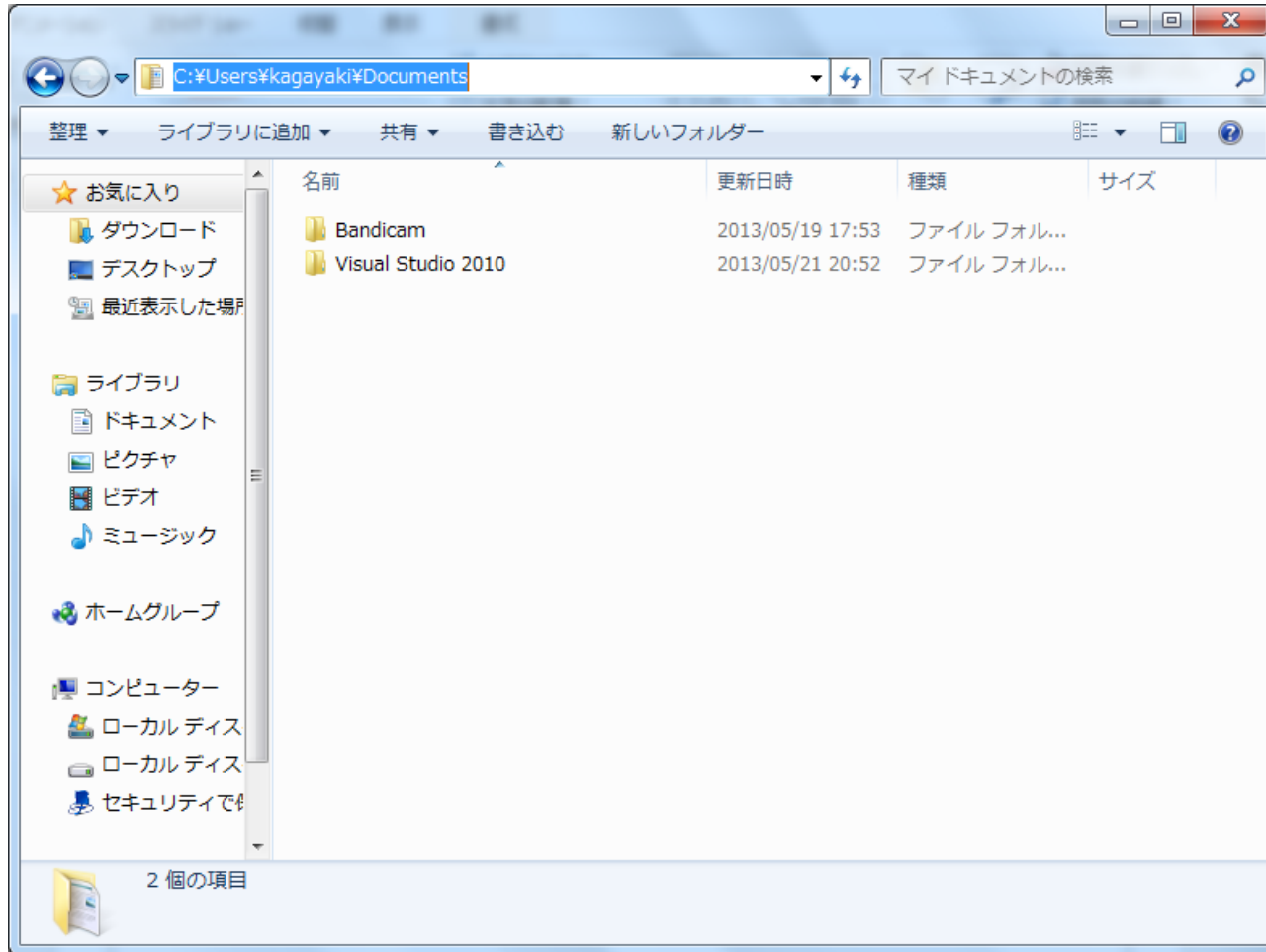
- こんな画面で



このあたりを
クリックすると、

ファイル入出力のプログラム例3/5

- パスが出てくるのでコピーしておく。




ファイル入出力のプログラム例4/5

```
#include<stdio.h>

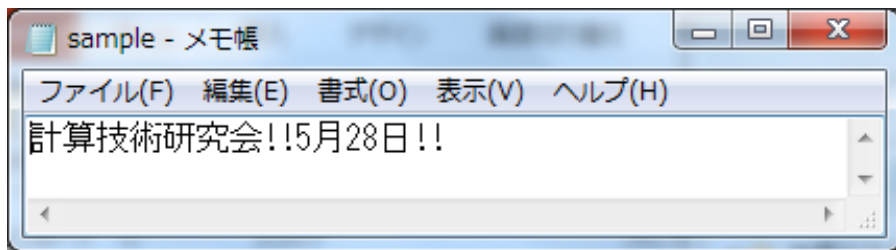
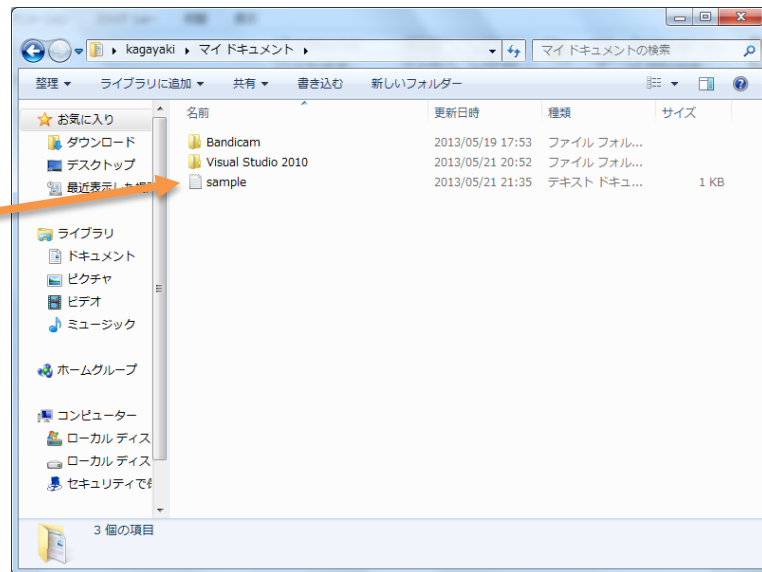
int main(){
    FILE *fp;
    fp=fopen("C:¥¥Users¥¥kagayaki¥¥Documents¥¥sample.txt","a");
    if(fp==NULL){
        printf("ファイルオープンに失敗しました¥n");
        return 0;
    }
    fprintf(fp,"計算技術研究会!!5月28日!!");
    fclose(fp);
    return 1;
}
```

コピーしたパスの、¥を2つにしたもの




ファイル入出力のプログラム例5/5

- デバッグなしで開始→実行結果には何も出ていないが...
- ドキュメントの中にsample.txtができています
- ダブルクリックして開くと、



今日の講座はこんなプログラムについてです。

ファイル入出力のプログラムの流れ

- ファイルポインタを定義する
 - FILE *fp;
 - ファイルポインタを初期化する “ ” を忘れずに!
 - fp=fopen(“(パス)¥¥(ファイル名.拡張子)”,”(モード)”);
 - ファイルポインタの初期化に失敗すると、
NULLが代入される
 - if(fp==NULL){
 printf(“ファイルのオープンに失敗しました¥n”);
 return 0;//main関数を抜けてプログラムを終了する
 }
 - 処理 - fprintf(~); など
 - ファイルポインタを解放する (約束事のようなもの)
 - fclose(fp);
- 

fopen関数の引数の「モード」について

- テキストファイル(文字とか数字とかのファイル)を扱う時

| モード | 意味 |
|-----|--------------------------|
| r | 読み込み専用:ファイルを編集できない |
| w | 書き込み専用:ファイルを編集できる |
| a | ファイルの最後に追加の書き込み |
| r+ | 読み込みと書き込み(ファイルがないとエラー) |
| w+ | 読み込みと書き込み(ファイルを作成する) |
| a+ | 読み込みと追加の書き込み(ファイルがないと作成) |

- バイナリファイル(画像とか音楽とかのファイル)を扱う時

| モード | 意味 |
|-----|-----------------|
| rb | バイナリモードで読み込み専用 |
| wb | バイナリモードで書き込み専用 |
| ab | バイナリモードで追加の書き込み |

問題1

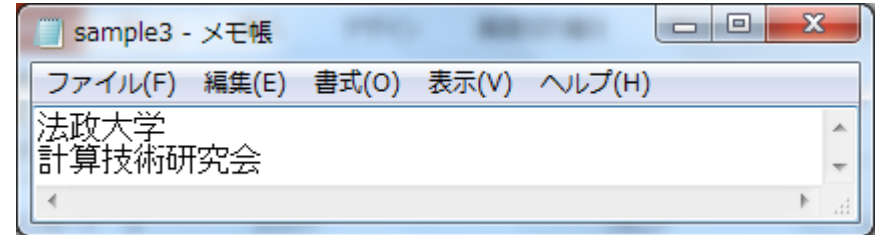
- デスクトップに、自分の名前が書いてある `sample2.txt` を プログラムを実行することによって、作成してください。ファイルポインタの初期化(ファイルのオープン)に失敗した場合も考慮すること。(必要に応じて6ページの例を見てください)
また、モードを変更してどういう挙動になるか見てください。
- 注意 最後のほうのページに答え

ファイル操作の標準関数

- fopen関数やfclose関数, fprintf関数
- fgetc(ファイルポインタ)
 - 先頭の一文字を得る。
 - 実行後はファイルポインタが次の一文字を指す。
 - …サンプルプログラムあります。
- fputc(char型の文字, ファイルポインタ)
 - 一文字書き込む。
 - 実行後はファイルポインタがひとつ先を指す。
 - …サンプルプログラムあります。

fgetc関数の例

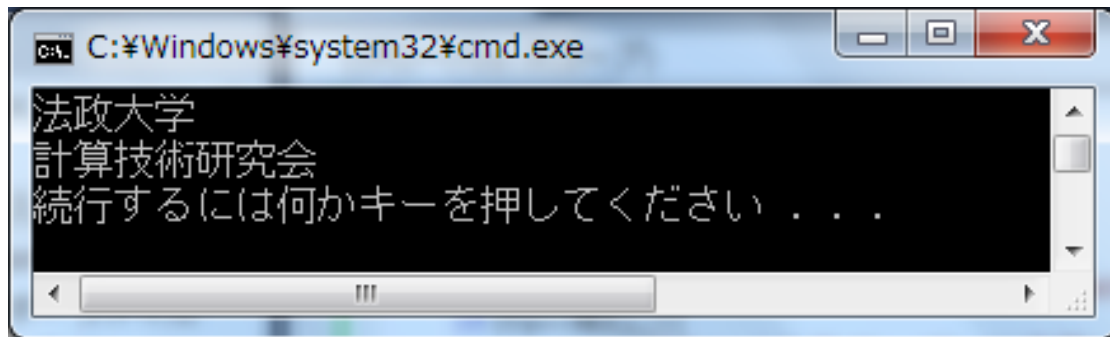
- あらかじめ、ドキュメントの中にsample3.txtを作成しておく



```
#include<stdio.h>
int main(){
    char c;
    FILE *fp;
    fp=fopen("C:¥¥Users¥¥kagayaki¥¥Documents¥¥sample3.txt","r");
    if(fp==NULL){
        printf("ファイルのオープンに失敗しました¥n");
        return 0;
    }
    while((c=fgetc(fp))!=EOF){
        printf("%c",c);
    }
    printf("¥n");
    fclose(fp);
    return 1;
}
```

各自のパス

実行結果



```
C:\Windows\system32\cmd.exe
法政大学
計算技術研究会
続行するには何かキーを押してください...
```

- sample3.txtファイルの中身が結果に表示される。
- sample3.txtから先頭の一文字読み取る
→それを表示→ファイルポインタが次の文字を指す
→ファイルポインタの指している一文字を読み取る
- ファイルポインタがEOFを指すまで上記を繰り返す。
- EOF (End Of File) はファイルの最後に必ずあるもの。
- fgetc関数はファイルポインタの指す一文字を返す。

fputc関数の例

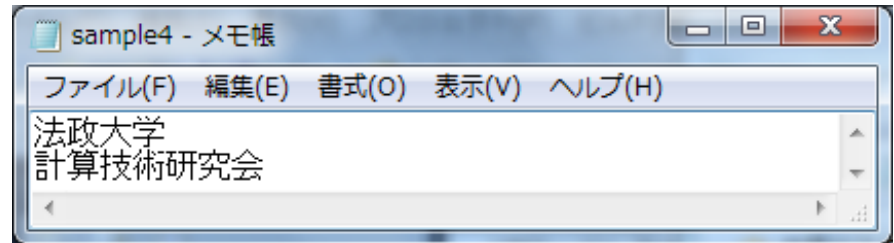
```
#include<stdio.h>

int main(){
    char c;
    FILE *fp,*cfp;
    fp=fopen("C:¥¥Users¥¥kagayaki¥¥Documents¥¥sample3.txt","r");
    cfp=fopen("C:¥¥Users¥¥kagayaki¥¥Documents¥¥sample4.txt","w");
    if(fp==NULL || cfp==NULL){
        printf("ファイルのオープンに失敗しました");
        return 0;
    }
    while((c=fgetc(fp))!=EOF){
        fputc(c,cfp);
    }
    fclose(fp);
    fclose(cfp);
    return 1;
}
```

fgetc関数を実行するとき作った
sample3があるパス

実行すると

- sample3のコピーのsample4が作成される。



- sample3からfgetc関数を実行→先頭の一文字を変数cに代入→fputc関数を使って、sample4に変数cの一文字を書き込む
- ファイルポインタfpがEOFを指すまで上記を繰り返す。
- fputcはファイルポインタの指している場所に一文字を書き込む


- 追加よてい

参考文献

- 参考文献:これならわかるC 入門の入門
– 坂下 夕里 著

問題1 回答例

モードを変更してどういう結果になるか見てみてください。
ファイルがない場合と、すでにある場合なども。



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

```
int main(){
```

```
    FILE *fp;
```

```
    fp=fopen("C:¥¥Users¥¥kagayaki¥¥Desktop¥¥sanple2.txt","w");
```

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

```
        printf("ファイルのオープンに失敗しました¥n");
```

```
        return 0;
```

```
    }
```

```
    fprintf(fp,"おさない じゅんぺい");
```

```
    fclose(fp);
```

```
    return 1;
```

```
}
```