

オペレーティングシステムの機能を使ってみよう 第3章 高水準入出力と低水準入出力

オペレーティングシステムの機能を使ってみよう

1 / 12

高水準入出力と低水準入出力

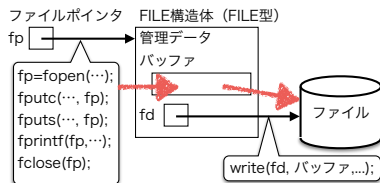
ファイルを読み書きするための機能
(API : Application Program Interface)

- 高水準入出力 (高水準 I/O)
多くの高機能な関数群
(fprintf(), fscanf(), fputc(), fgetc(), ...)
- 低水準出力 (高水準 I/O)
システムコールのこと
少なく、かつ、シンプルな API
(open(), read(), write(), lseek(), close())

オペレーティングシステムの機能を使ってみよう

2 / 12

高水準 I/O のデータ構造 (書き込み)

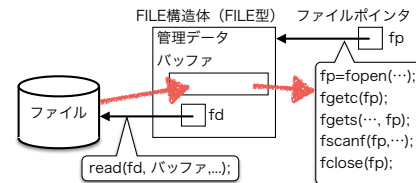


- ファイルポインタ (fp)
- FILE 構造体
- バッファリング
- write システムコール

オペレーティングシステムの機能を使ってみよう

3 / 12

高水準 I/O のデータ構造 (読み出し)



- ファイルポインタ (fp)
- FILE 構造体
- read システムコール
- バッファリング

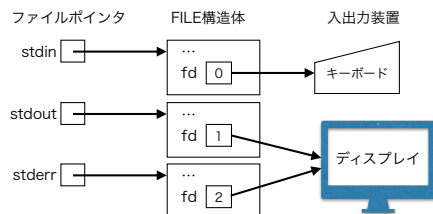
オペレーティングシステムの機能を使ってみよう

4 / 12

標準入出力 (標準入出力ストリーム)

名称	fd	fp	通常の接続先
標準入力ストリーム	0	stdin	キーボード
標準出力ストリーム	1	stdout	ディスプレイ
標準エラー出力ストリーム	2	stderr	ディスプレイ

fd : ファイルディスクリプタ
fp : ファイルポインタ



オペレーティングシステムの機能を使ってみよう

5 / 12

ユニファイド I/O

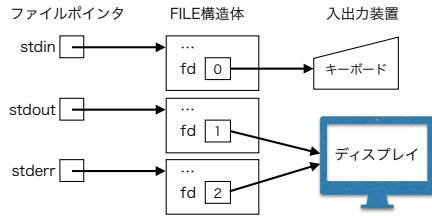
標準ストリーム	同じ意味の呼出し	役割
scanf(...)	fscanf(stdin, ...)	書式付きの入力
getchar()	fgetc(stdin)	1 文字入力
-	fgets(stdin, ...)	1 行入力
printf(...)	fprintf(stdout, ...)	書式付きの出力
putchar(c)	fputc(c, stdout)	1 文字出力
puts(buf)	fputs(buf, stdout)	1 行出力

- printf(...) と fprintf(stdout, ...) は同じ
- fp の代わりに stdin, stdout 等が使用できる。
- キーボードやディスプレイ (入出力装置) とファイルを同じ要領で操作できる。
- 入出力装置をファイルに統合 = (ユニファイド I/O)

オペレーティングシステムの機能を使ってみよう

6 / 12

標準入力ストリーム

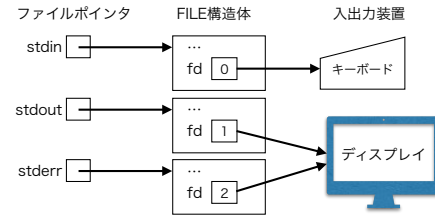


- ファイルポインタは stdin
- ファイルディスクリプタは 0 番
- ファイルディスクリプタ 0 は通常キーボードに接続
- ファイルポインタと FILE 構造体はプログラム起動時に初期化
- シェルはファイルディスクリプタ 0 をリダイレクト可能

オペレーティングシステムの機能を使ってみよ

7 / 12

標準出力ストリーム

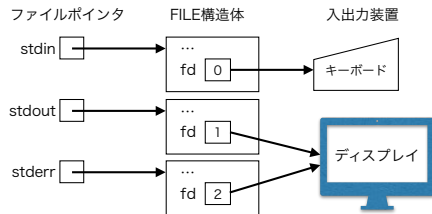


- ファイルポインタは stdout
- ファイルディスクリプタは 1 番
- ファイルディスクリプタ 1 は通常ディスプレイに接続
- ファイルポインタと FILE 構造体はプログラム起動時に初期化
- シェルはファイルディスクリプタ 1 をリダイレクト可能

オペレーティングシステムの機能を使ってみよ

8 / 12

標準エラー出力ストリーム



- エラーメッセージ出力用のストリーム
- ファイルポインタは stderr
- ファイルディスクリプタは 2 番
- ファイルディスクリプタ 2 は通常ディスプレイに接続
- ファイルポインタと FILE 構造体はプログラム起動時に初期化
- シェルはファイルディスクリプタ 2 をリダイレクト可能

オペレーティングシステムの機能を使ってみよ

9 / 12

性能比較 (1/2)

- 1 プログラムを準備する
 - mycp : 高水準 I/O 版
 - mycp2_1 : 低水準 I/O 版 (バッファサイズ=1 バイト)
 - mycp2_1024 : 低水準 I/O 版 (バッファサイズ=1,024 バイト)

- 2 大きめのファイルを作る

```
$ dd if=/dev/urandom of=aaa bs=1024 count=10240 <-- 10MiB のファイル aaa を作る
10240+0 records in
10240+0 records out
10485760 bytes transferred in 1.019062 secs (10289621 bytes/sec)
$ ls -l aaa
-rw-r--r-- 1 sigemura staff 10485760 Apr 15 17:35 aaa <-- できている
$
```

オペレーティングシステムの機能を使ってみよ

10 / 12

性能比較 (2/2)

3 実行時間を測定方法

```
$ rm bbb <--- 念のため bbb を消す
rm: bbb: No such file or directory
$ time ./mycp2_1 aaa bbb
real 1m31.664s
user 0m11.653s
sys 1m16.554s
$ cmp aaa bbb <--- コピー結果が正常かチェック
$
```

4 実行時間の測定

mycp2 1						
	1回目	2回目	3回目	4回目	5回目	平均
real	18.021	17.812	17.709	17.744	17.679	17.793
user	1.707	1.674	1.695	1.723	1.692	1.698
sys	16.253	16.096	15.977	15.976	15.935	16.047

オペレーティングシステムの機能を使ってみよ

11 / 12

課題 No.2 : 三つのプログラムの性能比較

上記の性能比較を実際に行う。提出物は以下の通りとする。

- 1 三つのプログラムについて実行結果を整理したもの
- 2 使用したプログラムのソースコード
- 3 感想・考察 (ソースコードの余白に記入する)

オペレーティングシステムの機能を使ってみよ

12 / 12