

ファイル入出力のための関数群

プログラムの中から、ファイル内のデータを読み書きするための関数群を紹介する。

(1) ファイルのオープン / クローズ関数

1. 整数 fd `openr(文字列 filepath)`
filepath で指定したファイルを、データの読み込みのために開く。fd はファイル識別子であり、以後、fd を指定して、ファイルからの読み込み等を行う。
2. 整数 fd `openw(文字列 filepath)`
filepath で指定したファイルを、データの書き込みのために開く。指定したファイルが既に存在しているならば、そのファイルはこれから書き出すデータで置き換えられる。
3. 整数 fd `opena(文字列 filepath)`
filepath で指定したファイルを、データの追記書き込みのために開く。指定したファイルが既に存在しているならば、そのファイルの続きとして、これから書き出すデータを追記する。
4. `close(整数 fd)`
fd で指定したファイルを閉じる。

(2) ファイルの読み書きのための関数

1. 文字列 str `getstr(整数 fd, 整数 n)`
fd で指定したファイルから、n 文字読み込んで その文字列を返す。ただし、改行コード (Return, Linefeed) もそれぞれ 1 文字と数える。
2. 文字列 str `getline(整数 fd)`
fd で指定したファイルから、1 行読み込んで その文字列を返す。
3. `putstr(整数 fd, 文字列 str)`
fd で指定したファイルに、str の文字列を書き出す。
4. `putline(整数 fd, 文字列 str)`
fd で指定したファイルに、str の文字列を書き出す。putstr() は、文字列だけを書き出すのに比べて、putline() は、文字列の後に改行コードも書き出す。

(3) ファイル処理のための補助関数

1. 文字列 str `isfile(文字列 filepath)`
filepath で指定したファイルが存在するかどうかを調べる。ファイルが存在する場合 "true"、存在しない場合 "false" の文字列が返される。
2. `rename(文字列 filepath1, 文字列 filepath2)`
filepath1 で指定したファイルの名前を、filepath2 に置き換える。
3. `flush(整数 fd)`
fd で指定されるファイルに書き込み途中のデータを強制的に書込む。(ファイルは通常バッファリングされるので、この関数を呼び出すことで、バッファ内容を吐き出す)
4. `remove(文字列 filepath)`
filepath で指定したファイルを削除する。