

Today: Reading and writing text files

Computer Programming 143 – Lecture 24

File Processing I

Faculty of Engineering



Featured Dataset

TMDB 5000 Movie Dataset

Metadata on ~5,000 movies from TMDb



The Movie Database (TMDb) • last updated 5 hours ago

779

Overview

Data

Kernels

Discussion

Activity

Download (9 MB)

New Kernel

2 Files (43.62 MB)

tmdb_5000_cred...

tmdb_5000_mov...

Download All

tmdb_5000_credits.csv

7.64 MB • Updated 6 hours ago

Download

About this file

Movie credits data

File handling functions

- fopen
- fclose
- fscanf
- rewind
- fgets
- fEOF
- fprintf

line 2 3 7 0 10

Extra:

fptr == NULL

Special files:

- stdout
- stderr
- stdin

```

#include <stdio.h>
#include <stdlib.h>
int main( void )
{
    FILE *fPtr; // file pointer
    char fileName[ 30 ]; // string to store file name
    char lineOfText[ 50 ] = "A single line of text";
    int myInt = 5;
    float myFloat = 1.345;
    // Next 2 lines illustrate a point - please do not use!
    fprintf( stdout, "Please enter filename: " );
    fscanf( stdin, "%29s", fileName ); //scanf( "%29s", fileName );
    fPtr = fopen( fileName, "w" ); // opens file for writing
    if ( fPtr == NULL ) {
        printf( "ERROR - File could not be opened!\n" );
    }
    else { // what would the following write to the file?
        fprintf( fPtr, "*" );
        fprintf( fPtr, "%s", lineOfText );
        fprintf( fPtr, "*" );
        fprintf( fPtr, "\nmyInt = %d\nmyFloat = %.3f\n", myInt, myFloat );
        fclose( fPtr ); // closes file
    }
    return 0; // indicates successful termination
} // end function main

```



```

#include <stdio.h>
#include <stdlib.h>
int main( void )
{
    FILE *fPtr; // file pointer
    char fileName[ 30 ]; // string to store file name
    char lineOfText[ 30 ], string1[ 20 ], string2[ 20 ];
    float myFloat;
    printf( "Please enter filename: " );
    scanf( "%29s", fileName ); // reads file name from user
    fPtr = fopen( fileName, "r" ); // opens file for reading
    if ( fPtr == NULL ) {
        printf( "ERROR - file could not be opened!\n" );
    }
    else { // what would the following code display?
        fgets( lineOfText, 29, fPtr );
        while ( !feof( fPtr ) ) {
            printf( "%s", lineOfText );
            fgets( lineOfText, 29, fPtr );
        }
        rewind( fPtr );
        printf( "%c\n", fgetc( fPtr ) );
        fgets( lineOfText, 29, fPtr );
        printf( " %s", lineOfText );
        fgets( lineOfText, 29, fPtr );
        printf( "%s", lineOfText );
        fscanf( fPtr, "%19s%19s%f", string1, string2, &myFloat );
        printf( "Value of myFloat = %.3f\n", myFloat );
        fclose( fPtr ); // closes file
    }
    return 0; // indicates successful termination
} // end main

```

Monday

Binary files

- Unformatted (stored as “raw bytes”)
 - Data not human readable
- Written and read sequentially or randomly

Today

Text files

- Data is stored as characters (**char**)
- Usually only written and read sequentially

File open modes

Mode	Description
r	Open a file for reading.
w	Create a file for writing. If the file already exists, discard the current contents.
a	Append; open or create a file for writing at end of file.
<u>r+</u>	Open a file for update (reading and writing).
w+	Create a file for update. If the file already exists, discard the current contents.
a+	Append; open or create a file for update; writing is done at the end of the file.
rb	Open a file for reading in binary mode.
wb	Create a file for writing in binary mode. If the file already exists, discard the current contents.
ab	Append; open or create a file for writing at end of file in binary mode.
rb+	Open a file for update (reading and writing) in binary mode.
wb+	Create a file for update in binary mode. If the file already exists, discard the current contents.
ab+	Append; open or create a file for update in binary mode; writing is done at the end of the file.