

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»
Факультет информационных технологий и программирования

Программирование
Лабораторная работа №4

Выполнил студент:	Шайдулин Михаил Андреевич
Группа:	М3106

Санкт-Петербург
2021

1 metaeditor.c

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <locale.h>

int main(int argc, char **argv)
{
    char* songname;
    char* prop_name;
    char* prop_value;
    uint8_t show;
    int get_flag = 0;
    int show_flag = 0;

    // for loop to initialize the parameters from terminal
    for (int argv_index = 1; argv_index < argc; argv_index++)
    {
        // file path parameter
        if (argv[argv_index][2] == 'f')
        {
            // path to a song
            songname = argv[argv_index] + 11;
            // code here
        }
        // show parameter
        else if (argv[argv_index][2] == 's')
        {
            show_flag = 1;
        }
        // get parameter
        else if (argv[argv_index][2] == 'g')
        {
            prop_name = argv[argv_index] + 6;
            get_flag = 1;
        }
        // set parameter
        else if (argv[argv_index][2] == 's')
        {
            prop_name = argv[argv_index] + 6;
            // code here
        }
        else if (argv[argv_index][2] == 'v')
        {
            prop_value = argv[argv_index] + 8;
            // code here
        }
    }

    // opening file.mp3
    FILE * music_file = fopen(songname, "rb+");
    if (music_file == NULL)
    {
        printf ("Error opening %s\n", songname);
        exit(1);
    }
    // initializing file size
    fseek (music_file , 0 , SEEK_END); // setting position to the end of the
    file
    long lSize = ftell (music_file); // getting size in bytes
    fseek (music_file , 0 , SEEK_SET); // setting pointer to the start of the
    file
    printf("%ld bytes\n", lSize); //
```

```

char signature[4];
char name_of_frame[5];
char artist[30];
char album[30];
char year[4];
char description[30];
unsigned short genre;

fread(signature, 1, sizeof(signature)-1, music_file); // reading signature
signature[3] = '\0'; // closing line

fseek(music_file, 3, SEEK_CUR);
int size;
char bsize[4];
fread(bsize, sizeof(bsize), 1, music_file);
size = ((bsize[0] << 21) | (bsize[1] << 14) | (bsize[2] << 7) | (bsize[3]));
printf("signature: %s\n\n", signature);

while (ftell(music_file) < size) {
    fread(name_of_frame, 1, sizeof(name_of_frame)-1, music_file); // reading
    in buffer
    name_of_frame[4] = '\0'; // closing line

    int size_of_frame;
    char fsize[4];
    fread(fsize, 1, sizeof(fsize), music_file); // reading
    size_of_frame = ((fsize[0] << 24) | (fsize[1] << 16) | (fsize[2] << 8) |
        (fsize[3]));
    fseek(music_file, 2, SEEK_CUR);
    int unicode = 0;
    unicode = fgetc(music_file);
    char * frame_content = (char *)malloc(size_of_frame);
    fread(frame_content, sizeof(char), size_of_frame-1, music_file);
    frame_content[size_of_frame-1] = '\0'; // closing line

    if (name_of_frame && !name_of_frame[0])
    {
        break;
    }
    if (show_flag == 1)
    {
        //printing content
        printf("frame: %s, %d bytes\n", name_of_frame, size);
        printf("content: %s\n\n", frame_content);
    } else if (get_flag == 1 && strcmp(name_of_frame, prop_name) == 0)
    {
        printf("frame: %s, %d bytes\n", name_of_frame, size);
        printf("content: %s\n\n", frame_content);
    }
}

// } else {
//     printf("frame_content: %s\n", frame_content);
// }
// shutting down
fclose (music_file);
return 0;
}

```