

WAV file concatenator

WAVE форматът е стандартизиран формат за аудио файлове разработен от IBM и Microsoft с цел съхранение на аудио данни. Линкове с повече информация:

<http://soundfile.sapp.org/doc/WaveFormat/>

<https://docs.fileformat.com/audio/wav/>

[https://en.wikipedia.org/wiki/WAV#:~:text=Waveform%20Audio%20File%20Format%20\(WAVE,Windows%20systems%20for%20uncompressed%20audio](https://en.wikipedia.org/wiki/WAV#:~:text=Waveform%20Audio%20File%20Format%20(WAVE,Windows%20systems%20for%20uncompressed%20audio)

Напишете програма на C, която обработва WAV файлове и:

1. Приема следните аргументи от командния ред

Опция	Значение	Задължителна	Аргумент	Ефект
s	silent	Не	Няма	Спира всички логове от програмата
n	numeral system	Не	dec/hex/oct	Определя бройната система

2. При стартиране, прочита от стандартния вход два пълни пътя съответно до два WAV файла, прочита съдържанието им и извежда извежда на стандартния изход информация за разликата в дължината на **звукото съдържание** в следния формат:

The sound content of <file-name-1> has <value> more bytes sound data than <file-name-2>

Стойността <value> се отпечатва в бройната система според подадения от командния ред аргумент или в десетична бройна система ако не е подаден.

3. Прочита от стандартния вход два символни низа, име на файл и позиция. Обединява звукото съдържание на двата файла в един, като съответно:

- Ако позицията е “front”, **звукото съдържание** на файла, чието име е въведено се поставя в предната част на новото звуково съдържание
- Ако позицията е “back”, **звукото съдържание** на файла, чието име е въведено се поставя в задната част на новото звуково съдържание
- Между двете звукови съдържания има 5 секунди тишина

Новото съдържание се записва в нов файл с име “името-на-първия-файл-името-на-втория-файл-concatenated.wav”

Примерни файлове за тестване на програмата може да вземете от тук: <https://mixkit.co/free-sound-effects/>

Допълнителни изисквания и насоки:

- Програмата спира изпълнение ако е подаден файл в друг формат или такъв с невалиден формат
- Използвайте подходящи типове данни – с правилни дължина и знак

- Обърнете внимание на подредбата на байтовете (little-endian vs. big-endian) в WAV хедъра
- Организирайте кода на програмата си в отделни файлове и функции
- Когато се очаква потребителят да въведе стойност от клавиатурата, изведете подходящо съобщение на стандартния изход
- Логовете, които отпечатва програмата на стандартния изход ясно показват от коя функция са, и кои са релевантните данни при изпълнението ѝ (аргументи, връщани стойности и т.н) изход
- За обработката на опциите и аргументите от командния ред използвайте функцията “getopt”

Бъдете готови да демонстрирате работата на програмата, както и да отговаряте на допълнителни въпроси. Възможно е от вас да бъде поискано да модифицирате части от програмата на място в деня на защитата.

При наличие на въпроси с удоволствие ще се радвам да ги обсъдим

Кристиян Златков

kristiqnz11@gmail.com