TALLINNA TEHNIKAÜLIKOOL

INFOTEHNOLOOGIA TEADUSKOND

Arvutitehnika instituut

Süsteemitarkvara õppetool

Henri Johann Norden 178945EAAB

IAX0583 Programmeerimine I

**HELIFAILIDE AUTOMAATNE KONVERTEERIMINE**

Projekt kodutöö asenduseks

Juhendaja: dotsent Vladimir Viies

Tallinn 2017

Autorideklaratsioon

Kinnitan, et käesolev töö on minu töö tulemus ja seda ei ole minu ega kellegi teise poolt varem esitatud.

Henri J. Norden

Sisukord

[Ülesande püstitus 4](#_Toc501416704)

[Programmi seletus 5](#_Toc501416705)

[common.h & common.c 5](#_Toc501416706)

[utils.h & utils.c 5](#_Toc501416707)

[loadsave.h & loadsave.c 6](#_Toc501416708)

[processor.h & processor.c 6](#_Toc501416709)

[main.c 6](#_Toc501416710)

[Pildid programmist 7](#_Toc501416711)

[Kasutatud kirjandus 9](#_Toc501416712)

# Ülesande püstitus

Koostada programm, mis:

1. käib rekursiivselt läbi kõik kasutaja valitud kausta alamkaustad
2. leiab kaustadest kõik helifailid ja konverteerib need kasutaja valitud parameetrite järgi
3. salvestab kaustastruktuuri, failide asukoha ja faile identifitseerivad andmed (suurus, loomis- ja muutmisaeg) mällu
4. salvestab failistruktuuri mälust püsivale andmekandjale
5. järgnevatel programmi kasutuskordadel suudab lugeda püsivale andmekandjale salvestatud failistruktuuri taas mällu ja tuvastada vahepeal failidega toimunud muudatusi
6. konverteerib muudetud failid uuesti ja salvestab muudatused
7. täiendavalt võiks konverteerimise kiirendamiseks suuta kasutada kõiki olemasolevaid protsessori tuumi

Helifailide konverteerimiseks on kasutuses dBpoweramp tarkvarapaketi käsurea põhine konverteerimisvahend. Programmi eeldatav kasutusplatvorm on Microsoft Windows, seega ühilduvus teiste operatsioonisüsteemidega ei ole oluline.

# Programmi seletus

Programm koosneb peamoodulist *main.c* ja neljast päisfailidega alammoodulist: *common*, *utils*, *loadsave* ja *processor*.   
Programmi lähtekood ja kompileeritud fail on koos aruandega samas *.zip* failis. Programm vajab korrektseks töötamiseks dBpoweramp tarkvara ja ilma selleta saab kasutajat vaid teavitada failimuudatustest.

## common.h & common.c

Failides common.h ja common.c on kirjeldatud kõik globaalsed muutujad, millest enamuse moodustavad kasutaja poolt muudetavad seaded. Siin on defineeritud ka funktsioon, mis loeb sisse programmile käsurealt antud argumendid ja seadistab programmi nende põhjal vastavalt. Failis common.h on kirjeldatud ka *struct*’e *file* ja *directory*, mida programm kasutab vastavalt failide ja kasutade mälus salvestamiseks. Antud juhul on tegemist *linked list* tüüpi andmehoidjatega.

## utils.h & utils.c

Moodulis *utils* on defineeritud erinevad abifunktsioonid:

* *readStr, readInt, readLInt, writeStr, writeInt, writeLInt* erinevate andmetüüpide vastavalt failist lugemiseks või faili kirjutamiseks,
* *longToTime, timeToLong* võimaldavad muuta Windowsi *FILETIME* andmetüübi tavaliseks täisarvuks ja vastupidi,
* *getExtension* tagastab funktsiooni argumendina edastatud failinimest faililaiendi,
* *spaces* tagastab tühikutest koosneva sõne, milles tühikute arv on võrdne globaalse muutuja *PROCESSING\_DEPTH* väärtusega,
* *addMessage* lisab sõnumi massiivi, millest need prinditakse konsooli alles siis, kui muud programmi tegevused on lõppenud,
* *getFilePath* valib failisüsteemi andmebaasile failinime,
* *buildCommand* paneb kokku käsu, mis süsteemi käsureal jooksutades konverteerib selle funktsiooni argumentidega kirjeldatud faili teise formaati.

Moodulis *utils* sisaldub moodul *common*.

## loadsave.h & loadsave.c

See moodul sisaldab peamisi funktsiooni failistruktuuri kõvakettalt lugemiseks ja salvestamiseks. Funktsioon *loadData* käib järjest läbi salvestatud kaustad, taastab kaustade andmed ja kutsub välja funktsiooni *readFile*, mis loeb andmebaasist mällu ühe faili andmed. Funktsioonid *saveData* ja *writeFile* teevad vastavalt vastupidist.

Moodulis *loadsave* sisaldub moodul *utils*.

## processor.h & processor.c

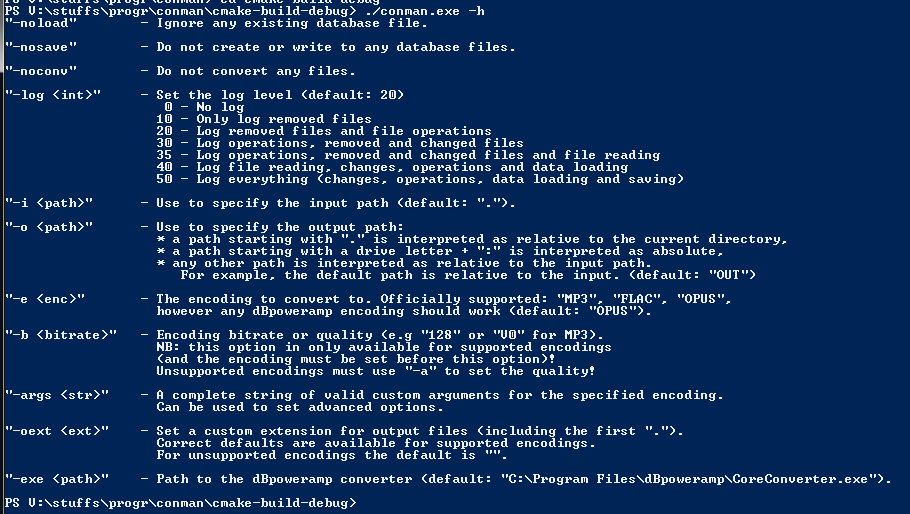
Tegemist on kõige massiivsema alammooduliga, mis tegeleb nii failistruktuuri rekursiivse läbimise, failide ja kasutade andmete parsimise ja helifailide konverteerimisega. Funktsioon *processDir* käib läbi kõik kaustad ja kontrollib funktsiooni *setFileData* abil, kas failide metaandmed on muutunud. Juhul kui on, siis funktsioon *runCommand* jooksutab esimesel vabaneval protsessori tuumal *buildCommand* funktsioonist saadud käsku. Lisaks kontrollib pärast funktsioon *removeDeletedFiles* kas on faile või kaustasid vahepeal ära kustutatud ja korrigeerib funktsioonidega *removeFile* ja *removeDir* andmebaasi vastavalt. Kasutajat teavitatakse ka programmi lõpus kustutatud failidest, et Tal oleks võimalik need varasemalt konverteeritud failide hulgast kergemini ära kustutada.

Moodulis *processor* sisaldub moodul *loadsave*.

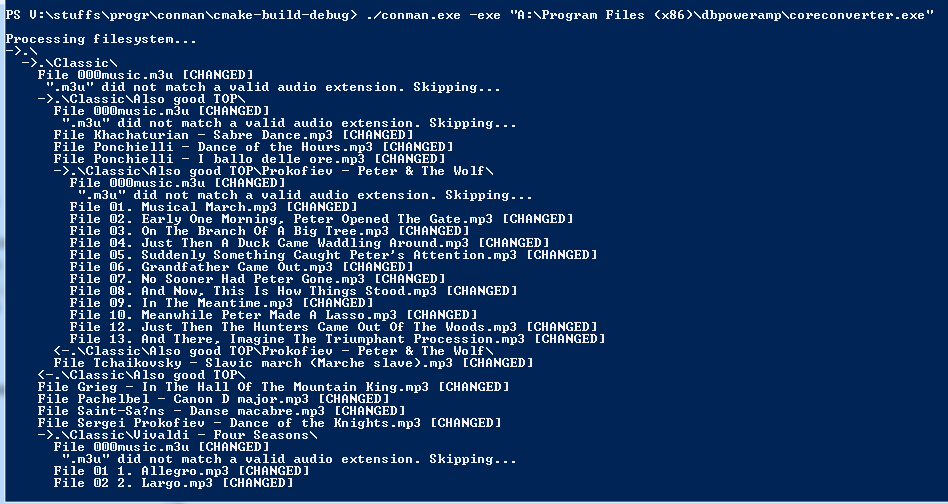
## main.c

Peamoodul seob kokku kõik programmi osad: programmile antud käsurea argumentide lugemise, andmete laadimise kõvakettalt, andmetöötluse ja uute andmete salvestamise tagasi kõvakettale. *Main* moodulis sisaldub moodul *processor*.

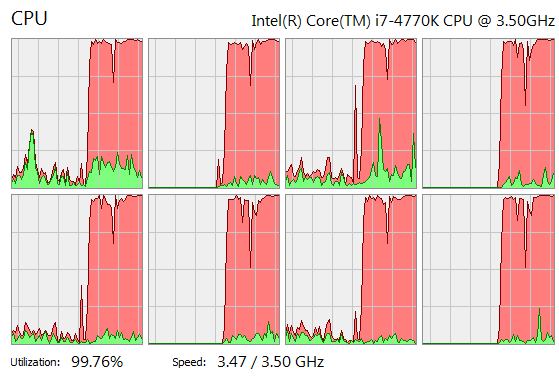
# Pildid programmist



Joonis . Programmi abiinfo, kus on näha programmis kasutaja poolt muudetavad parameetrid



Joonis . Programm konverteerides



Joonis . Programmi kasutab efektiivselt olemasolevat arvutusjõudlust konverteerimise ajal

# Kasutatud kirjandus

1. Microsoft Windows Dev Center <https://developer.microsoft.com/en-us/windows/desktop/develop>