

PRAKTIKUM IZ PROGRAMIRANJA 2

- domaći zadatak broj 5 -

Sastaviti program na programskom jeziku C kojim se vrši obrada tekst datoteka koje predstavljaju muzičke plejliste i konverzija tih datoteka iz jednog formata u drugi. Program treba da:

- Pročita jednu ili više zadatih ulaznih datoteka;
- Smesti podatke u jednu ili više odgovarajućih **dvostruko ulančanih** lista;
- Izvrši odgovarajuću obradu nad ulaznim podacima;
- Upiše rezultat u zadatu datoteku po tačno definisanom formatu u tekstu zadatka;
- Dealocira svu alociranu dinamičku memoriju i zatvori korišćene datoteke.

Zavisno od rednog broja problema, svaki student treba da sastavi **jedan** od programa koji su dati u prilogu ovog dokumenta.

Sa povećanjem procesorske snage sa jedne, i multimedijalnih mogućnosti računara sa druge strane, pojava zapisa sa visokim kvalitetom zvuka bila je više nego očekivana. Tržište medijuma je ispratilo nove potrebe tržišta računara novim formatima koji pružaju veliki kapacitet za smeštanje podataka (razne varijante CD, DVD, smeštanje u oblaku i slično). Razvoj različitih algoritama kompresije sa minimalnim gubicima je još više olakšao reprodukciju zvučnih zapisa na kućnim računarima. Zbog velike količine zapisa, potrebno je organizovati ih radi reprodukcije po željenom redosledu. Datoteka koja sadrži osnovne podatke o numerama koje treba reprodukovati i redosledu u žargonu se naziva **plejlista** (engl. *playlist*). Za potrebe ovog zadatka smatrati svaki **naziv** numere u okviru plejliste ima format **Ime izvođača – naziv numere**.

Postoji veći broj formata tekstualnih datoteka za smeštanje podataka o muzičkim plejlistama. Od interesa za rešavanje zadatka su tri često korišćena formata:

- M3U (<https://en.wikipedia.org/wiki/M3U>)
- PLS ([https://en.wikipedia.org/wiki/PLS_\(file_format\)](https://en.wikipedia.org/wiki/PLS_(file_format)))
- ASX (https://en.wikipedia.org/wiki/Advanced_Stream_Redirector)

Prikazani formati mogu imati i složeniju strukturu od one koja je predstavljena u okviru ovog zadatka, ali sve formate treba koristiti onako kako su ovde opisani. Sva tri tipa formata mogu biti napravljena uz pomoć programa KMPlayer, koji može biti preuzet sa adrese <http://kmplayer.en.softonic.com/>, dok M3U i PLS liste koristi i program Winamp, koji može biti preuzet sa adrese <http://www.winamp.com/>. Liste koje kreiraju KMPlayer ili Winamp se po formatu mogu minimalno razlikovati od formata koji je prikazan u ovom dokumentu. Navedeni programi se mogu koristiti prilikom testiranja rešenja.

M3U format

```
#EXTM3U
#EXTINF:302,Chris Rea - Looking For The Summer
C:\muzika\Chris Rea - Looking For The Summer.mp3
#EXTINF:265,Simply Red - Holding Back The Years
C:\muzika\Simply Red - Holding Back The Years.mp3
#EXTINF:207,Mambo Kings - Luz de luna
C:\muzika\Mambo Kings - Luz de luna.mp3
```

Prva linija M3U datoteke označava da se radi o proširenom formatu M3U liste i ona je ista u svakoj plejlisti. U nastavku je opis numera koje se nalaze u plejlisti. Svaka numera je predstavljena sa dva reda. Prvi red počinje sa oznakom #EXTINF:, nakon čega slede celi broj koji predstavlja trajanje numere u sekundama i naziv numere koji će biti prikazan u programu za reprodukciju. Druga linija predstavlja putanju do datoteke sa zvučnim zapisom na računaru.

PLS format

```
[playlist]
NumberOfEntries=3
File1=C:\muzika\Chris Rea - Looking For The Summer.mp3
Title1=Chris Rea - Looking For The Summer
Length1=302
File2=C:\muzika\Simply Red - Holding Back The Years.mp3
Title2=Simply Red - Holding Back The Years
Length2=265
File3=C:\muzika\Mambo Kings - Luz de luna.mp3
Title3=Mambo Kings - Luz de luna
Length3=207
Version=2
```

Prva linija PLS datoteke uvek ima sadržaj `[playlist]`. Druga linija govori koliko numera ima u plejlisti i ima format `NumberOfEntries=broj numera`. U nastavku sledi opis numera koje su u plejlisti. Svaka numera je opisana sa tri reda. Prvi ima format `File#=putanja do fajla`, drugi ima format `Title#=naziv numere`, a treći ima format `Length#=trajanje numere u sekundama`. Znak # označava redni broj numere koja se opisuje. Fajl se završava jednom linijom u formatu `Version=broj verzije`, koja govori o verziji PLS fajla (samo verzija 2 je značajna za ovaj zadatak).

ASX format

```
<Asx Version = "3.0" >

<Entry>
<Title>"Chris Rea - Looking For The Summer.mp3"</Title>
<Ref href = "C:\muzika\Chris Rea - Looking For The Summer.mp3"/>
</Entry>

<Entry>
<Title>"Simply Red - Holding Back The Years.mp3"</Title>
<Ref href = "C:\muzika\Simply Red - Holding Back The Years.mp3"/>
</Entry>

<Entry>
<Title>"Mambo Kings - Luz de luna.mp3"</Title>
<Ref href = "C:\muzika\Mambo Kings - Luz de luna.mp3"/>
</Entry>

</Asx>
```

ASX format je napisan na jeziku XML. Sve informacije o numerama su smeštene u tag `Asx` koji ima atribut `version` sa vrednošću 3.0. Informacije o numerama su smeštene u tag `<Entry>`. Naziv numere, koji će biti prikazan u plejeru, čuva se u tagu `Title` a putanja do fajla se čuva u tagu `Ref`, tačnije njegovom atributu `href`. Podaci o trajanju numere se ne čuvaju. Smatrati da se nakon svakog unosa nalazi tačno jedan prazan red, kao u primeru.

Navedene korake u izvršavanju programa realizovati kao zasebne funkcije prema rasporedu navedenom u tekstu svakog od zadataka. Po potrebi, mogu se realizovati i druge, pomoćne funkcije koje vrše deo obrade. Student sam treba da definiše imena funkcija, kao i argumente potrebne za njihov rad. Voditi računa da se funkcijama dostave samo neophodni podaci (pokazivač na početak liste i, po potrebi, podatke od kojih zavisi obrada). **Potprogrami ne smeju komunicirati pomoću globalnih promenljivih, već samo preko liste argumenata i povratne vrednosti.** Glavni program treba da poziva funkcije koje obavljaju gorenavedene radnje. Sve funkcije smestiti u odgovarajuće `.c` datoteke (spisku u napomeni), a prototipove svih funkcija smestiti u zajedničku `.h` datoteku.

U zavisnosti od rednog broja problema koji se rešava osmisliti sopstvenu strukturu ili strukture podataka za smeštanje podataka iz ulazne datoteke i rezultata. Element liste koja sadrži podatke iz ulazne datoteke treba realizovati kao **strukturu koja ima tačno tri polja – pokazivače na**

prethodni i sledeći element, i pokazivač na strukturu koja sadrži podatke. Voditi računa o pravilnoj alokaciji i dealokaciji dinamičke memorije.

Program sa korisnikom treba da interaguje putem komandne linije. Raspored i značenje pojedinačnih argumenata komandne linije su dati zasebno u okviru svakog zadatka.

U slučaju bilo kakve greške (poziv programa sa neodgovarajućim brojem ili vrstom argumenata komandne linije, neuspešna dodela dinamičke memorije, greška pri radu sa datotekom), ispisati poruku o grešci i korektno prekinuti izvršavanje (vraćanjem vrednosti 0 kao rezultata izvršavanja programa). U slučaju poziva programa sa neodgovarajućim brojem ili vrstom argumenata komandne linije, u programu ispisati poruku **ARG_GRESKA**. U slučaju neuspešne dodele dinamičke memorije, u programu ispisati poruku **MEM_GRESKA**. U slučaju greške pri radu sa datotekom, u programu ispisati poruku **DAT_GRESKA**. Zatim, korektno prekinuti izvršavanje programa (vraćanjem vrednosti 0 kao rezultata izvršavanja programa).

Ako nešto u postavci zadatka nije dovoljno precizno definisano ili ako su neki od zahteva međusobno suprotstavljeni, usvojiti razumnu pretpostavku i rešiti zadatak korišćenjem te pretpostavke. Na samoj odbrani obavestiti demonstratora o usvojenoj pretpostavci ili pretpostavkama. Programski kod rešenja zadatka treba da bude uredno komentaran, tako da pri pregledu programa lako može biti uočeno šta radi bilo koja programska celina. Takođe, treba poštovati pravila nazublivanja (indentacije) određenih celina prilikom pisanja koda.

Radi boljeg testiranja programa, odabrati nekoliko skupova podataka sa kojima će program biti testiran. Svaki primer treba da sadrži ulazne podatke i očekivani izlaz za te podatke.

Napomene:

1. Rok za predaju petog domaćeg zadatka je **utorak, 31.05.2022.** putem kursa predmeta na Moodle platformi za elektronsko učenje. Tačan termin za predaju će biti naknadno definisan za sve studente. Termin će biti ograničenog vremenskog trajanja.
2. Domaći zadaci će biti testirani i ocenjivani korišćenjem javnih i tajnih testova.
3. Studentima će nekoliko dana pre roka za predaju biti dostupno okruženje za testiranje rešenja domaćeg zadatka na Moodle platformi za elektronsko učenje korišćenjem javnih testova.
4. Prilikom predaje domaćeg zadatka studenti će rešavati i kratak test znanja u vezi rešenja domaćeg zadatka i relevantnog gradiva iz programskog jezika C koje obuhvata temu domaćeg zadatka. Na odbrani će biti postavljana pitanja, kao i dodatni zahtevi u vezi realizacije rešenja.
5. Domaći zadaci se rešavaju **samostalno**. Predmetni nastavnici zadržavaju pravo da nakon predaje domaćih zadataka izvrše proveru sličnosti i preduzmu odgovarajuće disciplinske mere.
6. Svi drugi detalji oko predaje, ocenjivanja i odbrane domaćeg zadatka će biti blagovremeno objavljeni.
7. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa): **$i = (R + G) \bmod 4$**
8. Kao rešenje domaćeg zadatka potrebno je predati na Moodle kursu predmeta sadržaj sledeće datoteke:
 - **dz5.c**, koja sadrži izvorni tekst kompletnog programa na programskom jeziku C;
9. Kao rešenje domaćeg zadatka potrebno je na odbrani napraviti sledeće datoteke:
 - **dz5.h**, koja sadrži prototipe svih funkcija opisanih u postavci zadatka;
 - **dz5.c**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku C;
 - **dz5_load.c**, **dz5_save.c**, **dz5_process.c**, koje sadrže izvorne tekstove funkcija potrebnih za inicijalizaciju programa, čitanje, snimanje i obradu kontakata;

0. Napisati program koji na osnovu dve zadate plejliste u PLS formatu kreira kreira novu plejlistu u ASX formatu koja sadrži samo muzičke numere istog izvođača koje se nalaze u obe zadate plejliste. Program najpre putem komandne linije prihvata imena ulaznih PLS datoteka. Nakon toga, program učitava podatke iz ulaznih datoteka, a potom formira rezultujuću plejlistu tako da su podaci sortirani leksikografski rastuće prema nazivu autora, a u slučaju istog autora prema imenu numere. Na kraju, program ispisuje rezultat obrade u odgovarajuću datoteku.

Program napisati prema sledećim stavkama.

- 1) Implementirati funkciju koja čita podatke o jednoj numeri iz ulazne datoteke prema formatu koji je zadat u tekstu zadatka i formira strukturu podataka kojom se opisuje jedna numera.
- 2) Implementirati funkciju koja pozivanjem prethodno realizovane funkcije formira dvostruko ulančanu listu numera.
- 3) Implementirati funkciju koja na osnovu dve zadate dvostruko ulančane liste numera formira rezultujuću dvostruko ulančanu listu numera koja sadrži samo muzičke numere istog izvođača koje se nalaze u obe zadate plejliste sortirane leksikografski rastuće prema nazivu autora, a u slučaju istog autora prema imenu numere.
- 4) Implementirati funkciju koja vrši formiranje izlazne datoteke `result.asx` na osnovu zadate rezultujuće dvostruko ulančane liste iz prethodne stavke.
- 5) Implementirati funkciju koja oslobađa svu zauzetu dinamičku memoriju.
- 6) Napisati glavni program koji korišćenjem prethodno realizovanih funkcija pročita ulazne datoteke, izvrši zahtevanu obradu, formira izlaznu datoteku, dealocira svu korišćenu dinamičku memoriju i zatvori korišćene datoteke.

Primeri

Primer 1	
Poziv programa:	<code>program.exe 1st.pls 2nd.pls</code>
Ulazna datoteka:	<code>1st.pls 2nd.pls</code>
Izlazna datoteka:	<code>result.asx</code>
Standardni izlaz:	-
Primer 2	
Poziv programa:	<code>program.exe 1st.pls 2nd.pls</code>
Ulazna datoteka:	<code>1st.pls 2nd.pls</code>
Izlazna datoteka:	<code>result.asx</code>
Standardni izlaz:	-
Primer 3	
Poziv programa:	<code>program.exe</code>
Ulazna datoteka:	-
Izlazna datoteka:	-
Standardni izlaz:	<code>ARG_GRESKA</code>

1. Napisati program koji na osnovu zadate plejliste u M3U formatu kreira novu plejlistu u PLS formatu koja sadrži maksimalno N najkraćih numera svakog izvođača iz originalne plejliste. Program najpre putem komandne linije prihvata ime ulazne M3U datoteke, izlazne PLS datoteke i broj N. Nakon toga, program treba da učitava podatke iz ulazne datoteke u dvostruko ulančanu listu. Potom, program treba da sortira formiranu listu leksikografski rastuće po imenu izvođača, a numere istog izvođača rastuće po vremenu trajanja. Ukoliko dve numere nekog izvođača imaju isto vreme trajanja, sortirati ih leksikografski rastuće po naslovu numere. Na kraju, program za svakog izvođača treba da izdvoji maksimalno N najkraćih numera i upiše ih u izlaznu datoteku.

Program napisati prema sledećim stavkama.

- 1) Implementirati funkciju koja čita podatke o jednoj numeri iz ulazne datoteke prema formatu koji je zadat u tekstu zadatka i formira strukturu podataka kojom se opisuje jedna numera.
- 2) Implementirati funkciju koja pozivanjem prethodno realizovane funkcije formira dvostruko ulančanu listu numera.
- 3) Implementirati funkciju koja sortira zadata dvostruko ulančanu listu po prethodno navednom kriterijumu.
- 4) Implementirati funkciju koja iz sortirane dvostruko ulančane liste izbacuje sve nepotrebne numere.
- 5) Implementirati funkciju koja vrši formiranje izlazne PLS datoteke na osnovu zadate rezultujuće dvostruko ulančane liste iz prethodne stavke.
- 6) Implementirati funkciju koja oslobađa svu zauzetu dinamičku memoriju.
- 7) Napisati glavni program koji korišćenjem prethodno realizovanih funkcija pročita ulaznu datoteku, izvrši zahtevanu obradu, formira izlaznu datoteku, dealocira svu korišćenu dinamičku memoriju i zatvori korišćene datoteke.

Primeri

Primer 1	
Poziv programa:	<code>program.exe in.m3u out.pls 10</code>
Ulazna datoteka:	<code>in.m3u</code>
Izlazna datoteka:	<code>out.pls</code>
Standardni izlaz:	-
Primer 2	
Poziv programa:	<code>program.exe in.m3u out.pls 21</code>
Ulazna datoteka:	<code>in.m3u</code>
Izlazna datoteka:	<code>out.pls</code>
Standardni izlaz:	-
Primer 3	
Poziv programa:	<code>program.exe</code>
Ulazna datoteka:	-
Izlazna datoteka:	-
Standardni izlaz:	<code>ARG_GRESKA</code>

2. Napisati program koji na osnovu zadate plejliste u PLS formatu kreira novu plejlistu u M3U formatu koja sadrži samo muzičke numere čije je trajanje kraće ili jednako zadatom vremenu sortirane po imenu izvođača i nazivu numere. Program najpre putem komandne linije prihvata ime ulazne PLS datoteke, izlazne M3U datoteke i maksimalno vreme trajanja numere u sekundama. Nakon toga, program učitava podatke iz ulazne datoteke, a potom formira rezultujuću plejlistu tako da su podaci sortirani leksikografski rastuće prema imenu izvođača, a u slučaju istog izvođača leksikografski rastuće prema nazivu numere. Na kraju, program ispisuje rezultat obrade u odgovarajuću datoteku.

Program napisati prema sledećim stavkama.

- 1) Implementirati funkciju koja čita podatke o jednoj numeri iz ulazne datoteke prema formatu koji je zadat u tekstu zadatka i formira strukturu podataka kojom se opisuje jedna numera.
- 2) Implementirati funkciju koja pozivanjem prethodno realizovane funkcije formira dvostruko ulančanu listu numera.
- 3) Implementirati funkciju koja na osnovu zadate dvostruko ulančane liste numera formira rezultujuću dvostruko ulančanu listu numera koja sadrži samo muzičke numere čije je trajanje kraće ili jednako zadatom vremenu, sortirane leksikografski rastuće prema imenu izvođača, a u slučaju istog izvođača leksikografski rastuće prema nazivu numere.
- 4) Implementirati funkciju koja vrši formiranje izlazne M3U datoteke na osnovu zadate rezultujuće dvostruko ulančane liste iz prethodne stavke.
- 5) Implementirati funkciju koja oslobađa svu zauzetu dinamičku memoriju.
- 6) Napisati glavni program koji korišćenjem prethodno realizovanih funkcija pročita ulaznu datoteku, izvrši zahtevanu obradu, formira izlaznu datoteku, dealocira svu korišćenu dinamičku memoriju i zatvori korišćene datoteke.

Primeri

Primer 1	
Poziv programa:	<code>program.exe plejlista.pls plejlista.m3u 300</code>
Ulazna datoteka:	<code>plejlista.pls</code>
Izlazna datoteka:	<code>plejlista.m3u</code>
Standardni izlaz:	-
Primer 2	
Poziv programa:	<code>program.exe ulaz.pls izlaz.m3u 320</code>
Ulazna datoteka:	<code>ulaz.pls</code>
Izlazna datoteka:	<code>izlaz.m3u</code>
Standardni izlaz:	-
Primer 3	
Poziv programa:	<code>program.exe</code>
Ulazna datoteka:	-
Izlazna datoteka:	-
Standardni izlaz:	<code>ARG_GRESKA</code>

3. Napisati program koji kreira nekoliko plejsita u M3U formatu na osnovu jedne plejliste koja je zadata u ASX formatu. Program putem komandne linije prihvata ime ulazne ASX datoteke. Program zatim učitava podatke iz ulazne datoteke i sortira rastuće sve numere prvo po imenu izvođača, a zatim po imenu numere. Nakon toga, program za svakog izvođača formira jednu datoteku u M3U formatu (dužina svake pesme se postavlja na vrednost 0).

Program napisati prema sledećim stavkama.

- 1) Implementirati funkciju koja čita podatke o jednoj numeri iz ulazne datoteke prema formatu koji je zadat u tekstu zadatka i formira strukturu podataka kojom se opisuje jedna numera. Ukoliko ne pročita numeru, funkcija treba da vrati vrednost `NULL`.
- 2) Implementirati funkciju koja pozivanjem prethodno realizovane funkcije formira dvostruko ulančanu listu numera.
- 3) Implementirati funkciju koja sortira dvostruko ulančanu listu, prema zahtevima zadatka.
- 4) Implementirati funkciju koja vrši formiranje izlaznih datoteka na osnovu dvostruko ulančane liste. Ime jedne izlazne datoteke se formira dodavanjem ekstenzije `.m3u` na ime izvođača čije pesme se nalaze u toj datoteci.
- 5) Implementirati funkciju koja oslobađa svu zauzetu dinamičku memoriju.
- 6) Napisati glavni program koji korišćenjem prethodno realizovanih funkcija pročita ulaznu datoteku, izvrši zahtevanu obradu, formira izlaznu datoteku, dealocira svu korišćenu dinamičku memoriju i zatvori korišćene datoteke.

Primeri

Primer 1	
Poziv programa:	<code>program.exe lista.asx</code>
Ulazna datoteka:	<code>lista.asx</code>
Izlazna datoteka:	<code>Chris Rea.m3u Mambo Kings.m3u Simply Red.m3u</code>
Standardni izlaz:	-
Primer 2	
Poziv programa:	<code>program.exe lista2.asx</code>
Ulazna datoteka:	<code>lista2.asx</code>
Izlazna datoteka:	<code>Chris Rea.m3u Mambo Kings.m3u Simply Red.m3u</code>
Standardni izlaz:	-
Primer 3	
Poziv programa:	<code>program.exe</code>
Ulazna datoteka:	-
Izlazna datoteka:	-
Standardni izlaz:	<code>ARG_GRESKA</code>