

Praktikum iz programiranja 2

Materijal potreban za peti domaći zadatak

Opširnija objašnjenja o svim postojećim formatima plej-listi (engl. *playlist*) se mogu pronaći uz pomoć Internet pretraživača.

U ovom dokumentu je priložen kratak i nepotpun opis tri formata za predstavljanje plej-listi, kao i primer plej-liste istog sadržaja u tri različita formata.

Priložena su i uputstva za korišćenje funkcija standardne biblioteke programskog jezika C za očitavanje i ispisivanje proteklog i trenutnog vremena.

Sadržaj

Ukratko o plej-listama	2
Formati plej-listi	2
M3U	2
PLS	2
ASX	3
Funkcije programskog jezika C za rad sa vremenom	3
Često postavljana pitanja	3

Ukratko o plej-listama

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 i slično). Razvoj raznih formata 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 **plej-lista** (engl. *playlist*).

Formati plej-listi

Prikazani formati mogu imati i složeniju strukturu od one koja je predstavljena ovde, ali sve formate treba koristiti onako kako su ovde opisani. Više informacija o prikazanim formatima možete naći na:

<http://en.wikipedia.org/wiki/M3U>

[http://en.wikipedia.org/wiki/PLS_\(file_format\)](http://en.wikipedia.org/wiki/PLS_(file_format))

http://en.wikipedia.org/wiki/Advanced_Stream_Redirector

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 fajlu. Zato će možda biti potrebno, pre testiranja programa sa rešenjem DZ5, nekim editorom teksta prilagoditi plej-listu formatu opisanom u ovom dokumentu.

M3U

```
#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 označava da se radi o proširenom formatu M3U liste i ona je ista u svakoj plej-listi. U nastavku je opis numera koje se nalaze u plejlisti. Svaka numera je predstavljena sa dva reda. Prvi red počinje sa **#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.

PLS

```
[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 uvek ima sadržaj **[playlist]**. Druga linija govori koliko numera ima u plej-listi i ima format **NumberOfEntries=broj numera**. U nastavku sledi opis numera koje su u plej-listi. 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 DZ5).

ASX

```
<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 numeri 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**.

Funkcije programskog jezika C za rad sa vremenom

Operativni sistem vodi evidenciju o utrošenom procesorskom vremenu za svaki od trenutno aktivnih procesa. U programskom jeziku C, ovo vreme se može dobiti pozivom funkcije `clock()` i izraženo je u „otkucajima“ sistemskog časovnika (engl. *clock ticks*). Svaka implementacija jezika C podrazumeva sopstvenu vrednost simboličke konstante `CLOCKS_PER_SEC` (u ranijim verzijama standarda `CLK_TCK`), koja predstavlja broj „otkucaja“ sistemskog časovnika u toku jedne sekunde.

Sa druge strane, u svakom trenutku je moguće od operativnog sistema dobiti informaciju o vremenu u sistemskom časovniku, kako i o datumu i vremenskoj zoni gde se dati računar nalazi. Jednom kada tačno vreme bude očitano, ne postoji nikakava prepreka da se ono pretvori u odgovarajuću strukturu, odnosno prikaže na željeni način preko proizvoljnog znakovnog niza. Priloženi programski segment ilustruje prikaz tačnog vremena bez prikaza datuma:

```
#include <time.h>
...
time_t pIntTime = time(NULL);
struct tm* currentLocalTime = localtime(&pIntTime);
char* dateTimeString = calloc(100+1, sizeof(char));
/* if time setting and memory allocation was succesfull, create complete message */
if (currentLocalTime && dateTimeString)
    /* format the time as needed */
    strftime(dateTimeString, 100, "%H:%M:%S", currentLocalTime);
```

Dodatne informacije o ovde navedenim funkcijama (i svim ostalim koje se nalaze u zaglavlju `time.h`) se mogu naći u zbirci prof. Krausa i na raznim dokumentacionim stranicama (kako onima koje dolaze uz razvojno okruženje, tako onima dostupnim na Internetu).

Često postavljana pitanja

Q: Šta su ulazni parametri programa?

A: Program dobija imena datoteka preko parametara komandne linije. Ove parametre korisnik zadaje razvojnom okruženju **Visual C++ 6.0** na sledeći način: stavka **Project** iz glavnog menija, zatim stavka **Settings** (ili **Alt+F7**). Dijalog koji će biti prikazan ima, između ostalih, i karticu **Debug**. Kategorija **General**, koja će standardno biti prva otvorena, ima mesto predviđeno za argumente komandne linije - **Program Arguments**. Tu treba uneti sve ono sto bi se u komandnoj liniji otkucalo

posle imena programa. Na primer, ako imamo "ulaz.m3u izlaz.pls", to znači da je ulazna datoteka `ulaz.m3u` i da je izlazna datoteka `izlaz.pls`. Slično je i u okruženju **Visual C++ 2008**: stavka **Project** iz glavnog menija, zatim stavka **Properties** (ili **Alt+F7**). Dijalog koji će biti prikazan ima sa leve strane kontrolu sa raznim stavkama. Treba izabrati stavku **Configuration Properties** i unutar nje **Debugging**. Sa desne strane će biti prikazano nekoliko opcija, od kojih je druga **Command Arguments** i ima istu namenu kao **Program Arguments** u **Visual C++ 6.0**.

Q: Kako da obezbedim uslovno prevođenje? Koristim `#ifdef`, `#elif` i sve ostale potrebne direktive, ali kako da definišem željeni simbol, a da ne koristim `#define PERA` koje moram da brišem kada ne želim da **PERA** bude definisano?

A:

Visual C++ 6.0: Prvo iz menija **Build** odaberite stavku **Configurations**, te u datom dijalogu dodajte (**Add...**) novu konfiguraciju baziranu na postojećoj `win 32 Debug` konfiguraciji, i dajte joj ime po želji. Zatvorite dijalog i idite na **Project->Settings** (**Alt+F7**). Tamo odaberite konfiguraciju koju želite da menjate, a zatim idite na karticu **C/C++**. Kategorija **General** sadrži stavku **Preprocessor Definitions**, gde treba dodati željeni simbol koji ste u tekstu programa koristili u `#ifdef` ili `#ifndef` direktivi preprocesoru. Kada želite da prevedete program korišćenjem date konfiguracije, koja sadrži ili ne sadrži odgovarajuće simbole, iz menija **Build** odaberite stavku **Set Active Configuration** i tu podesite željenu konfiguraciju.

Visual C++ 2005: Prvo iz menija **Build** odaberite stavku **Configuration Manager**, te u datom dijalogu izaberite listbox kontrolu koja pokazuje postojeću `Debug` konfiguraciju i ispod postojećih konfiguracija pronađite opciju `<New...>` i izaberite je. U sledećem dijalogu novoj konfiguraciji dajte ime po želji i obavezno u listbox kontroli `Copy settings from` izaberite postojeću `Debug` konfiguraciju. Zatvorite dijalog i idite na **Project->Properties** (**Alt+F7**). Tamo odaberite konfiguraciju koju želite da menjate. Dijalog koji će biti prikazan ima sa leve strane kontrolu sa raznim stavkama. Treba izabrati stavku **Configuration Properties** i unutar nje **C/C++**, unutar koje postoji **Preprocessor**. a desne strane će biti prikazano nekoliko opcija, od kojih je prva **Preprocessor Definitions** i ima istu namenu kao u **Visual C++ 6.0**. Tu treba dodati željeni simbol koji ste u tekstu programa koristili u `#ifdef` ili `#ifndef` direktivi preprocesoru. Kada želite da prevedete program korišćenjem date konfiguracije, koja sadrži ili ne sadrži odgovarajuće simbole, u **Build-> Configuration Manager** odaberite željenu konfiguraciju.