**ECE 270**



Justin Newman

Quiz #13

CD List

November 17, 2014

# Statement of the Problem

The purpose of this program is to parse a .txt file that contains a database of favorite albums of the class. The file is laid out with each line of the file representing an album and for each album, the artist name, album name, year released, genre, rating (specified by student who submitted it), and the name of the student who submitted the album.

Each piece of information for a given album is separated by a single ‘,’ . As the file is parsed, each album should be stored as a structure that has the 6 pieces of information regarding the album as elements.

# Description of solution

The program is incomplete, thus far the program parses the input file MasterCDList.txt using the strtok() function to separate each line into the individual pieces of information for each cd and stores them correctly in a structure array. The program also prints the structure array, though I have found no good way of formatting the output predictably. For the year and rating elements, the strings read in must be converted to integers and floating point numbers respectively, using the functions atoi() and atof().

The only other complete part of the program is the function that allows you to input your own values for the information on a single cd.

# Output and Testing

Once the program was working correctly it was simple to test the parsing and output, just compare the output to the MasterCDList.txt file to ensure they match.

As stated before, the output formatting is unpredictable at best:

Carly Rae Jepson Call Me Maybe-Single 2012 Pop 5.0 Alex Pappas

Bryan Adams The Best of Me 2002 Rock 5.0 Alex Pappas

Falco Falco Greatest Hits 1999 Techno 2.4 Alex Pappas

Green Day American Idiot 2004 Punk 3.4 Alex Pappas

Imagine Dragons Night Visions 2012 Alternative 5.0 Alex Pappas

Mitch Leigh Man of La Mancha 1973 Musical 4.0 Tim Hollman

Paul O'Neill Trans-Siberian Orchestra Beethoven's Last Night 2000 Symphonic Rock 4.5 Tim Hollman

The Tallest Man on Earth Shallow Grave 2011 Folk 4.8 Tim Hollman

Jon Por Birgisson Go 2010 Post-Rock 4.3 Tim Hollman

--------------------------Output Truncated------------------------------------

Native Deen My Faith My Voice 2011 Rap 5.0 Abdulaziz Abuzir

Native Deen I Am Near 2011 Rap 5.0 Abdulaziz Abuzir

Native Deen Mercy to Mankind 2011 Rap 5.0 Abdulaziz Abuzir

Native Deen Small Deeds 2005 Rap 5.0 Abdulaziz Abuzir

Native Deen Zamilooni 2005 Rap 5.0 Abdulaziz Abuzir

The Black Eyed Peas Light up the Night 2010 Pop 5.0 Bashar Yousif

The Black Eyed Peas Boom Boom Pow 2009 Pop 5.0 Bashar Yousif

The Black Eyed Peas Imma Be 2009 Pop 5.0 Bashar Yousif

The Black Eyed Peas I Goota Feeling 2009 Pop 5.0 Bashar Yousif

The Black Eyed Peas Whenever 2009 Pop 5.0 Bashar Yousif

Linkin Park Hybrid 2000 Nu Metal 5.0 Jordan Stepchuk

Styx Styx Greatest Hits 1995 Rock 5.0 Jordan Stepchuk

The Avett Brothers I and Love and You 2009 Folk Rock 4.0 Jordan Stepchuk

3 Doors Down 3 Doors Down 2008 Alternative Rock 4.0 Jordan Stepchuk

Three Days Grace Three Days Grace 2003 Alternative Metal 3.5 Jordan Stepchuk

# Code

1 #include<stdio.h>

2 #include<stdlib.h>

3 #include<string.h>

4

5 **struct** cd{

6 **char** artistName[50];

7 **char** albumName[50];

8 **int** year;

9 **char** genre[50];

10 **float** rating;

11 **char** studentName[50];

12 };

13 **int** fillStruct(FILE \*,**struct** cd []);

14 **void** setCD(**struct** cd \*);

15 **void** printStructArray(**struct** cd [],**int** );

16 **int** main()

17 {

18 FILE \*fp;

19 fp=fopen("MasterCDList.txt","r");

20 **int** numLines=0;

21 **struct** cd CDList[175];

22 **struct** cd MyCD;

23 numLines=fillStruct(fp,CDList);

24 setCD(&MyCD);

25 printStructArray(CDList,numLines);

26 fclose(fp);

27 **return** 0;

28 }

29 **int** fillStruct(FILE \*fp,**struct** cd CDList[175])

30 {

31 **int** i=0;

32 **char** \*token;

33 **char** year[5], rating[4], temp[100], comma[2]=",\0";

34

35 **for**(i=0;i<175;i++)

36 {

37 //printf("error\n %d",i); - testing

38

39 fgets(temp,**sizeof**(temp),fp);

40 token=strtok(temp,comma);

41 strcpy(CDList[i].artistName,strdup(token));

42

43 token=strtok(NULL,comma);

44 strcpy(CDList[i].albumName,strdup(token));

45

46 token=strtok(NULL,comma);

47 strcpy(year,strdup(token));

48 CDList[i].year=atoi(year);

49

50 token=strtok(NULL,comma);

51 strcpy(CDList[i].genre,strdup(token));

52

53 token=strtok(NULL,comma);

54 strcpy(rating,strdup(token));

55 CDList[i].rating=atof(rating);

56

57 token=strtok(NULL,comma);

58 strcpy(CDList[i].studentName,strdup(token));

59

60 }

61

62 **return**(i);

63 }

64

65 **void** setCD(**struct** cd \*MyCD)

66 {

67 **char** year[5], rating[4];

68 printf("Enter the CD's artist: ");

69 gets(MyCD->artistName);

70 printf("Enter the album name: ");

71 gets(MyCD->albumName);

72 printf("enter the year the album was released: ");

73 gets(year);

74 MyCD->year=atoi(year);

75 printf("Enter the album's genre: ");

76 gets(MyCD->genre);

77 printf("Enter your rating (0.0-5.0): ");

78 gets(rating);

79 MyCD->rating=atof(rating);

80 printf("Enter your name: ");

81 gets(MyCD->studentName);

82 printf("%s",MyCD->artistName);

83 }

84 **void** printStructArray(**struct** cd CDList[],**int** n)

85 {

86 **int** i;

87 printf("\n\tArtist\t \tAlbum\t \tYear\t \tGenre\t \tRating\t \tStudent\t");

88 **for**(i=0;i<175;i++)

89 {

90 FILE \*fout;

91 fout=fopen("quiz13.txt","w");

92 printf("%s",CDList[i].artistName);

93 printf("\t\t%s\t",CDList[i].albumName);

94 printf("\t%d",CDList[i].year);

95 printf("\t%s",CDList[i].genre);

96 printf("\t%.1f",CDList[i].rating);

97 printf("\t%s",CDList[i].studentName);

98 printf("\n");

99 fprintf(fout,"%s \t",CDList[i].artistName);

100 fprintf(fout,"%s \t",CDList[i].albumName);

101 fprintf(fout,"%d \t",CDList[i].year);

102 fprintf(fout,"%s \t",CDList[i].genre);

103 fprintf(fout,"%.1f \t",CDList[i].rating);

104 fprintf(fout,"%s \t",CDList[i].studentName);

105 fprintf(fout,"\n");

106 fclose(fout);

107 }

108 }