CENG 313 – Operating Systems
Homework #2

Implement a C program that is explained below using Unix. You should implement a multi-threaded program that operates on linked lists that contain music genres. Your program requires three types of threads:

  A. Cutting and pasting threads.
  B. Disliked song removing threads.
  C. Old song removing threads.

In your program you will be given two types linked list.

> **Linked list type 1 (playlist(s)):** Traversing the list and cutting the song(s).

> **Linked list type 2 (userplaylist):** Paste the song(s) to this common list.

Type "**A**" threads should use two linked list (one "**Linked list type 1**" type list and one "**Linked list type 2**" type list) at the same time. Without having both of "lists" thread cannot perform its task. Type "**A**" thread should cut and paste a song at each time. Each type "**A**" thread should decide the total number of music it's going to cut/paste randomly. The random decision shouldn't be less then 1 and greater than 10.

Type "**B**" thread is responsible of cleaning the disliked song genres in the playlist.

Type "**C**" thread is responsible of cleaning the songs older than a certain year in the playlist.

These lists could be **"Linked list type 1" or "Linked list type 2"** type of list. Therefore, thread types "**B**" and "**C**" can use "**Linked list type 1**" and" **Linked list type 2**" at any time. However, these threads should consume only one resource at a time.

NOTE 1: It is important that "**Linked list type 2**" is going to be the list that has all the contents of "**Linked list type 1**" type list and at the end all the "**Linked list type 1** type list should be empty.

The amount of list types is given below;

  • Linked List Type 1 = **2 lists (playlist1 and playlist2)**
  • Linked List Type 2 = **1 list (userplaylist)**

The amount of thread types is given below;

  • Thread Type A = **3 threads**
  • Thread Type B = **1 threads**
  • Thread Type C **= 1 threads**

NOTE 2: Your program should initialize the 2 playlists with random songs (At least 10 songs for each list). **Userplaylist** is initially empty. Please use data structure for song specified below.

Your program should get the disliked genre and year (the user wants to delete all songs before this year) from user. When the work with the lists is finished, print the contents of all lists to the console.

struct song {

        char name [25];

        char genre [15];

        int year;//(1992)

}

NOTE 3: You must prevent any possible deadlocks and starvation in this homework and please also recall that this is like a Readers/Writers problem. Therefore, make sure that you follow synchronization methodologies (i.e. semaphore, mutex).
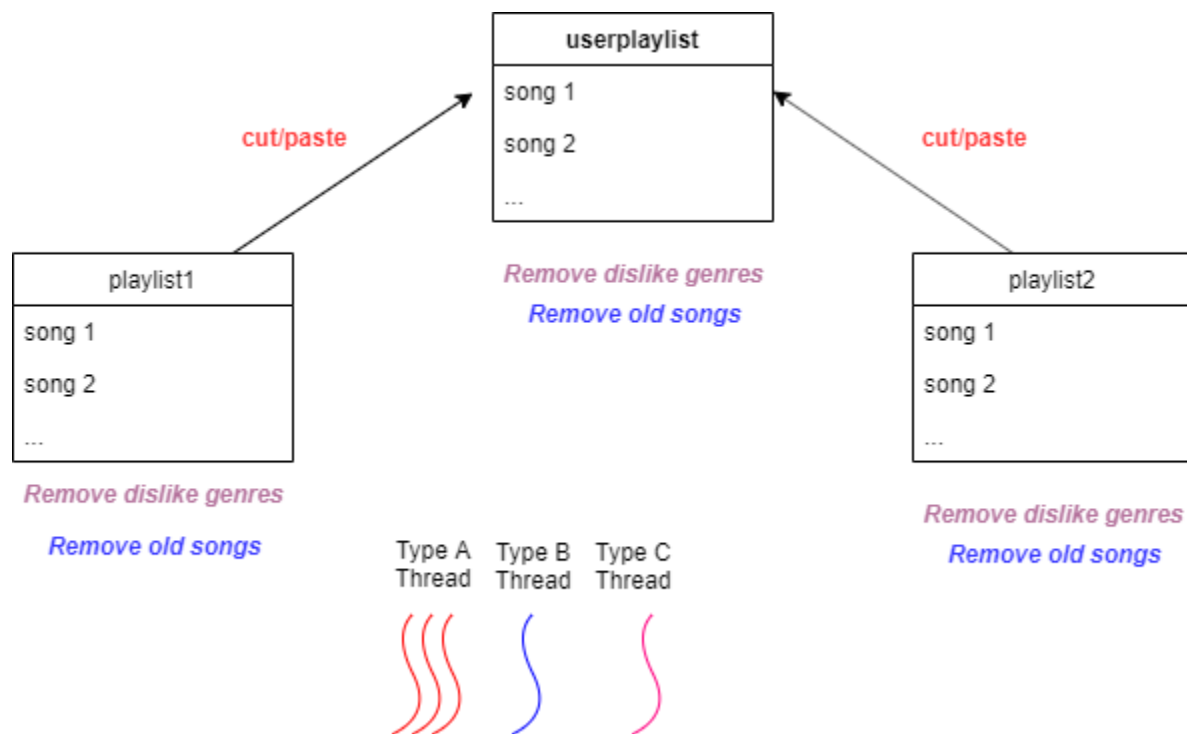
ASSIGNMENT RULES!

- Cheating will **NOT** be tolerated!
- For any detected cheating will be **graded as 0**.
- Late Submissions will not be allowed.

GRADE REDUCTIONS

        Since you are Junior students you are expected that you are aware of; error handlings, controls, software design etc. This lecture should be taken seriously and will take a crucial part in your work lives. Please code your programs wisely. Possible grade reductions,

- Lack of comment usage!

- Missing controls!
- No error handling!
- Unused/dead codes!
- Naming conventions!

Please do not discuss with us why your grades decreased just because you have done the programming sins listed above!

NOTE: Do not ask from us about the possible errors that could occur. From this lecture and labs, you are expected to be aware of the possible errors.