

Project Report on

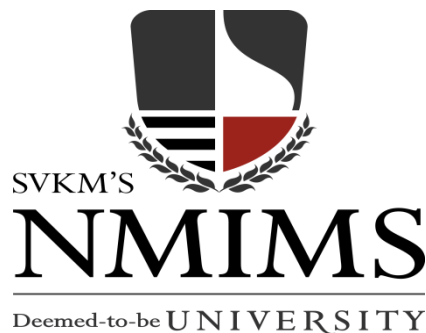
I Music

By

N064 Bhavyaraj Singh
N069 Anshuman Thapliyal
N078 Khushi Vora
N079 Siddharth Phadtare

Under the Guidance of
Pankti Doshi

Department of Computer Engineering
Mukesh Patel School of Technology, Management, and Engineering



MUKESH PATEL SCHOOL OF TECHNOLOGY
MANAGEMENT & ENGINEERING
SVKM's

NARSEE MONJEE INSTITUTE OF MANAGEMENT STUDIES (Declared as Deemed-to-be
University Under Section 3 of the UGC Act, 1956)

V. L. Mehta Road, Vile Parle (West)

MUMBAI -400056

March 2020

Subject: Data Structures

Semester III, Year: II

Academic Year: 2021-2022

Index

Sr. No.	Topic	Page No.
1	Introduction	3
2	Implementation Code	5
3	Screenshots	22
4	Function Description	
5	Conclusion	

Introduction

Description:

Music is a part of every person's life. No matter what your mood is, you have a song to sustain that mood. **I Music is a music player app.** The code that we have developed allows the users to play the songs present either from the Music Library or create their own Personalized playlist. The user can access the Music Library or their Playlist via the Main Menu.

Features present in I Music:

1. Main Menu

- (i) Music Library
- (ii) Playlist
- (iii) Exit from the app

2. Music Library Menu

- (i) Insert
- (ii) Display all songs
- (iii) Search by title
- (iv) Insert into playlist
- (v) Song on repeat
- (vi) Playlist Menu

3. Playlist Menu

- (i) Insert
- (ii) Display playlist
- (iii) Total duration of playlist
- (iv) Search by title
- (v) Shuffle
- (vi) Swapping using title
- (vii) Delete by Sr No.

Data Structure:

We will be using **Singly Link List** for the implementation of the code

Reason:

We could have used stack and queue data structure using array instead of singly linked list data structure. Following are the reason for discarding stack and queue.

Stack & queue	Singly Link List
Static data structure / Dense list	Dynamic data structure
Continuous memory allocation	Dynamic memory allocation
Size of the arrays is pre-defined	Size of the singly link list is not pre-defined
Inserting/deletion a new element is expensive because the room has to be created/deleted for the new elements	Efficient memory utilization as the memory is not continuous

The playlist is being implemented using a singly linked list as each node in the linked list represents a song, and it has several data members(title, artist and duration) corresponding to various attributes of the song.

Code

```
#include <stdio.h>
#include <stdlib.h>

struct playlistNode //Singly Linked List for Playlist
{
    char song_name[10];
    char artist_name[20];
    long int duration;
    struct playlistNode *next1;
}*start1 = NULL;//Starting address of playlistNode List

struct libraryNode //Singly Linked List for Music Library
{
    char song_name[10];
    char artist_name[20];
    long int duration;
    struct libraryNode *next2;
}*start2 = NULL;//Starting address of libraryNode List

void insert_end(char[], char[], long int); //Insertion into music library
void traverse_library(); //Traversal through music library
void search_by_title_in_library(); //To search in the music library using title
void insert_into_playlist(); //To insert a pre-existing song from music library into
playlist
void traverse_playlist(); //Traversal through playlist
void show_duration_of_the_playlist(); //Calculating the total duration of playlist
void search_by_title_in_playlist(); //To search in the playlist using title
void swap_using_songname_in_a_playlist(); //Swapping positions of two songs in a
playlist using their title
void shuffle(int); //To shuffle the playlist
void remove_a_song_from_the_playlist(int); //To remove a song from the playlist using
it's Sr No.
void repeat_song(); //To play a song on loop

int main()
{
    int choice, x = 1, choices, m = 1, position, q = 1, choice1,l=0;
    long int duration1;
    char songname[10], artistname[20];
    printf("\n");
```

```

printf("\n O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O ");
printf("\n * * ");
printf("\n * * ");
printf("\n * WELCOME TO * ");
printf("\n * * ");
printf("\n * | I MUSIC | * ");
printf("\n * * ");
printf("\n * * ");
printf("\n O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O ");
printf("\n\n");

```

C: //Displaying choices for Main Menu

```

printf("\n\t\t** MAIN MENU **\n\tPRESS 1: Go to Music Library\n\tPRESS 2: Go
to Playlist\n\tPRESS 3: Exit from App\n\n"); //Functions for the Main Menu

```

```

while(q)
{
    printf("Enter the choice: ");
    scanf("%d", &choice1);

```

switch(choice1) //User choice for Main Menu

```

{
    case 1:
    {
        goto A;
        break;
    }

```

```

    case 2:
    {
        goto B;
        break;
    }

```

```

    case 3:
    {

```

```

printf("\n O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O ");
printf("\n * * ");
printf("\n * THANK YOU FOR USING OUR APP * ");
printf("\n * * ");
printf("\n O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O ");

```

```

return 0;
break;

```

```

    }

    default:
    {
        printf("\t\tInvalid Choice\n");
        printf("\t\tPlease choose again\n\n\n");
        goto C;
        break;
    }
}
}

```

A: //Displaying choice for Music Library

```

printf("\n\n\n\t\t*Menue for MUSIC LIBRARY*\n\tPress 1: Insert songs into the
Main Library\n\tPress 2: Display all songs in Main Library\n\tPress 3: Search by
Title\n\tPress 4: Insert song in Playlist\n\tPress 5: Enter to put song on Repeat\n\tPress
6: Go to the Playlist Menu\n\tPress 7: Return to Main Menu\n\n");

```

```

while(x)
{
    printf("\nEnter choice: ");
    scanf("%d", &choice);

    switch(choice) //User choice for Music Library
    {
        case 1:
        {
            l++;          // Counter to generate Sr No.
            printf("Sr No.: ");
            printf("%d\n",l); //printing auto-generated Sr No.
            printf("Song Name: ");
            scanf("%s",&songname);
            printf("Artist Name: ");
            scanf("%s",&artistname);
            printf("Duration(in seconds): ");
            scanf("%ld",&duration1);
            insert_end(songname, artistname, duration1);
            goto A;
            break;
        }

        case 2:
        {

```

```

        traverse_library();
        goto A;
        break;
    }

case 3:
{
    search_by_title_in_library();
    goto A;
    break;
}

case 4:
{
    insert_into_playlist();
    goto A;
    break;
}

case 5:
{
    repeat_song();
    goto A;
    break;
}

case 6:
{

```

```

    B: //Displaying choice for Playlist
    printf("\n\t\t**FOR PLAYLIST**\n");
    printf("\n\n\t\t * Menue for PLAYLIST * \n\tPress 1: Insert a song into
playlist\n\tPress 2: Display the playlist\n\tPress 3: Display total duration\n\tPress 4:
Search by Title\n\tPress 5: Shuffle playlist\n\tPress 6: Swap\n\tPress 7: Delete a song
from playlist\n\tPress 8: Return to Music Library\n\tPress 9: Return to Main
Menu\n\n");
    while(m)
    {
        printf("\nEnter choice: ");
        scanf("%d", &choices);

        switch(choices) //User choices for Playlist
        {
            case 1:
            {

```



```

        insert_into_playlist();
        goto B;
        break;
    }
case 2:
{
    traverse_playlist();
    goto B;
    break;
}

case 3:
{
    show_duration_of_the_playlist();
    goto B;
    break;
}

case 4:
{
    search_by_title_in_playlist();
    goto B;
    break;
}

case 5:
{
    shuffle(l);
    goto B;
    break;
}
case 6:
{
    swap_using_songname_in_a_playlist();
    goto B;
    break;
}

case 7:
{
    printf("Enter the Sr No. of the song to be delete:\n");
    scanf("%d", &position); //Input Sr No. of song to be deleted
    remove_a_song_from_the_playlist(position);
    goto B;
}

```

```

        break;
    }

    case 8:
    {
        goto A;
        break;
    }

    case 9:
    {
        goto C;
        break;
    }

    default:
    {
        printf("\t\tInvalid Choice\n");
        printf("\t\tPlease choose again\n\n\n");
        goto B;
        break;
    }
}

}

}

case 7:
{
    goto C;
    break;
}

default:
{
    printf("\t\tInvalid Choice.\n");
    printf("\t\tPlease choose again\n\n\n");
    goto A;
    break;
}
}

}

```

```

void insert_end(char songname[], char artistname[], long int duration1)
{
    struct libraryNode *newnode,*temp; //creating a newnode and temp pointer of the
    libraryNode list
    newnode = (struct libraryNode*)malloc(sizeof(struct libraryNode)); //allocating a
    memory for newnode pointer of the libraryNode list
    strcpy(newnode->song_name, songname); //copying the song name into newnode
    pointer
    strcpy(newnode->artist_name, artistname); //copying the artist name into newnode
    pointer
    newnode->duration = duration1; //assigning song duration into newnode pointer
    newnode->next2 = NULL;

    if(start2 == NULL) //to check if the libraryNode list is empty
    start2 = newnode; //storing the starting address

    else
    {
        temp = start2;
        while(temp->next2 != NULL) //loop will be executed till it reaches last node of the
        libraryNode list
        {
            temp = temp->next2; //the address of the next node will be assigned
        }

        temp->next2 = newnode; //the address of the newnode pointer will be assigned to the
        next temp pointer
    }
    printf("\nInsertion of Song Successful\n\n");
}

void traverse_library()
{
    int f=0;
    struct libraryNode*temp; //creating a temp pointer to traverse the libraryNode
    temp = start2; //assigning the address of the start pointer of the libraryNode to the
    temp pointer of the libraryNode

    if(temp == NULL) //to check if the list is empty
    {
        printf("Library is empty.\n");
        printf("Try again please\n\n\n");
    }
}

```

```

else
{
    printf("    Sr No. \t    Song Name\t    Artist Name\t    Duration(in
seconds)\n");
    while (temp!=NULL) //loop will execute till it reaches the last node of the
libraryNode list
    {
        f++; //Counter for Sr No.
        printf("\t %d \t\t %s\t\t %s\t\t %ld\n" ,f, temp->song_name, temp->artist_name,
temp->duration);
        temp = temp->next2; //assign the address of the next of current temp pointer to
the current temp pointer
    }
}

void search_by_title_in_library()
{
    int c = 1,f=0;
    char n[20];
    printf("Enter the Song Name to be searched: ");
    scanf("%s", &n); //Entering the song name to be searched
    struct libraryNode*temp; //creating a temp pointer of the libraryNode list
    temp = start2; //assigning the address of the start pointer of the libraryNode list to the
temp pointer of the libraryNode list

    if(start2 == NULL) //to check if the list is empty
    {
        printf("Library is empty.\n");
        printf("Please try again \n\n");
    }

    else
    {
        printf("    Sr No. \t    Song Name\t    Artist Name\t    Duration(in seconds)\n");
        while(temp!= NULL)
        {
            c = strcmp(n, temp->song_name); //to compare the input song name with the
search song name
            if(c==0) //if both the song names are same
            {
                f++;

```

```

        printf("\t %d \t\t %s\t\t %s\t\t %ld\n" ,f, temp->song_name, temp-
>artist_name, temp->duration);
        break;
    }
    temp = temp->next2;
}

if(c!=0) //if song name not found
{
    printf("-----\n");
    printf("Data not available.\n");
    printf("Please try again \n\n");
}
}

}

void insert_into_playlist()
{
    struct playlistNode*newnode, *temp1;
    newnode = (struct playlistNode*)malloc(sizeof(struct playlistNode));
    temp1 = start1; //assigning the address of the start1 pointer of the playlistNode list to
the temp1 pointer of the playlistNode list
    int c = 1, counter = 0;
    char n[20];
    printf("Enter the song title to be added to the playlist: ");
    scanf("%s", &n); //Enter the song name to be added to the playlist

    while(temp1!=NULL)
    {
        if(!strcmp(temp1->song_name, n)) //comparing the input song name with the song
name of the current node
        {
            counter++; //incrementing the counter
        }

        temp1 = temp1->next1; //assigning the address of the next of current temp pointer
to the current node pointer
    }

    if(counter == 0) //to check if the input song name is different than the song names in
the playlistNode list
    {
        struct libraryNode*temp;

```

```

temp = start2;

if(start2 == NULL)
{
printf("Library is empty.\n");
printf("Please try again \n\n");
return;
}

else
{
printf(" Song Name\t Artist Name\t Duration(in seconds)\n");
while(temp!= NULL)
{
c = strcmp(n, temp->song_name); //comparing the input song name with the
song name of the current node
if(c==0) //if they both are the same
{
printf("\t%s\t\t %s\t\t%ld\n" ,temp->song_name, temp->artist_name, temp-
>duration);
break;
}
temp = temp->next2;
}

if(c!=0) //if they both are different
{
printf("Data not available.\n");
printf("Please try again \n\n");
return;
}
}

strcpy(newnode->song_name, temp->song_name);
strcpy(newnode->artist_name, temp->artist_name);
newnode->duration = temp->duration;
newnode->next1 = NULL;

if(start1 == NULL)
start1 = newnode;

else
{

```

```

        newnode->next1 = start1;
        start1 = newnode;
    }
    printf("\n Insertion of song successful\n\n");
}

else
{
    printf("\nData is already in the playlist\n");
    return;
}

}

void traverse_playlist()
{
    struct playlistNode*temp;
    temp = start1; //assigning the address of the start pointer of the playlistNode list to the
temp pointer of the playlistNode list
    int f=0;

    if(temp == NULL)
    {
        printf("Playlist is empty.\n");
        printf("Please try again \n\n");
    }

    else
    {
        printf("    Sr No. \t Song Name\t Artist Name\t Duration(in seconds)\n");
        while (temp!=NULL)
        {
            f++;
            printf("\t %d \t\t %s\t\t %s\t\t %ld\n" ,f, temp->song_name, temp->artist_name,
temp->duration);
            temp = temp->next1;
        }
    }
}

void show_duration_of_the_playlist()
{
    struct playlistNode *temp;

```

```

long int sum = 0, n;

temp = start1;

if(temp == NULL)
{
    printf("Playlist is empty.\n");
    printf("Please try again \n\n");
}

else
{
    while(temp!=NULL)
    {
        n = temp->duration; //assigning the duration of the current node to n
        sum = sum + n; // adding the new duration of song to the pre-existing total
        duration
        temp = temp->next1;
    }
}

printf("\nPlaylist Duration(in seconds): %ld sec\n", sum); //printing the total duration
of the playlistNode list
}

void search_by_title_in_playlist()
{
    int c = 1,f=0;
    char n[20];
    printf("Enter the song name to be searched: ");
    scanf("%s", &n);
    struct playlistNode*temp;
    temp = start1;
    if(start1 == NULL)
    {
        printf("Playlist is empty.\n");
        printf("Please try again \n\n");
    }

    else
    {
        printf("    Sr No. \t Song Name\t Artist Name\t Duration(in seconds)\n");
        while(temp!= NULL)
        {

```



```

        c = strcmp(n, temp->song_name); //comparing the input song name with the
song name of the current node
        if(c==0) //if they both are same
        {
            f++; //incrementing the counter
            printf("\t %d \t\t %s\t\t %s\t\t %ld\n" ,f, temp->song_name, temp->artist_name,
temp->duration);
            break;
        }
        temp = temp->next1;
    }

    if(c!=0) //if song not found
    {
        printf("-----\n");
        printf("Data not available.\n");
        printf("Please try again \n\n");
    }
}
}

```

```

void swap_using_songname_in_a_playlist()
{
    char n1[10], n2[10];

    printf("\nBefore Swapping:\n\n");
    traverse_playlist(); //printing the playlistNode list before swapping

    struct playlistNode*temp1, *temp2, *temp3; //creating three temporary pointers:
temp1, temp2 and temp3 of the playlistNode list
    temp1 = start1;
    temp2 = start1; //assigning the starting node address to temp1 and temp2
    temp3 = (struct playlistNode*)malloc(sizeof(struct playlistNode)); //allocating
memory for temp3 pointer of the playlistNode list

    printf("Enter the song names to be swapped:\n");
    scanf("%s %s", n1, n2);

    if(temp1 == NULL) // to check if the playlistNode list is empty
    {
        printf("Playlist is empty.\n");
        printf("Try again please\n\n\n");
    }
}

```

```

    }

    else
    {
        while (temp1!=NULL) //loop will execute till temp1 pointer reaches the last node
of the playlistNode list
        {
            if(!strcmp(temp1->song_name, n1)) //if both the song names are same
            {
                while (temp2!=NULL) //it will keep on executing till temp2 pointer reaches
the last node of the playlistNode list
                {
                    if(!strcmp(temp2->song_name, n2)) //if both the song names are same
                    {
                        strcpy(temp3->song_name, temp2->song_name); // swapping the
information of the two selected nodes
                        strcpy(temp3->artist_name, temp2->artist_name);
                        temp3->duration = temp2->duration;

                        strcpy(temp2->song_name, temp1->song_name);
                        strcpy(temp2->artist_name, temp1->artist_name);
                        temp2->duration = temp1->duration;

                        strcpy(temp1->song_name, temp3->song_name);
                        strcpy(temp1->artist_name, temp3->artist_name);
                        temp1->duration = temp3->duration;

                    }

                    temp2 = temp2->next1;
                }
            }

            temp1 = temp1->next1;
        }
    }

    printf("\n\nAfter Swapping:\n\n");
    traverse_playlist(); //traversing the playlistNode list after swapping
}

void remove_a_song_from_the_playlist(int position)
{

```

```

    struct playlistNode *temp1, *temp2; //creating temp1 and temp2 pointers for the
playlistNode list
    int i;
    temp1 = start1;
    temp2 = start1; //asssigning the address of the starting node to temp1 and temp2
pointers of the playlistNode list
    if(start1 == NULL) //to check if the playlistNode list is empty
    {
        printf("Playlist is empty.\n");
        printf("Please try again \n\n");
    }
    else
    {
        for(i = 1; i<position - 1; i++) //to find the position entered by the user
        {
            temp1 = temp1->next1;
            temp2 = temp2->next1;
        }

        temp2 = temp2->next1;
        temp1->next1 = temp2->next1;
        free(temp2); //deleting the node from the playlistNode list
    }
    printf("\nDeletion successful\n");
    printf("New playlist\n");
    traverse_playlist(); //traversing the playlistNode list after deletion
}

void repeat_song()
{
    int c = 1, i = 1, k = 1, f = 0;
    char n[20];
    printf("Enter the song name to put on loop: ");
    scanf("%s", &n);
    printf("Number of times you want to loop a song : ");
    scanf("%d", &k);
    struct libraryNode *temp;
    temp = start2;
    if(start2 == NULL) //to check if the libraryNode list is empty
    {
        printf("Library is empty.\n");
        printf("Please try again \n\n");
    }
}

```

```

else
{
    printf("    Loop No. \t Song Name\t Artist Name\t Duration(in seconds)\n");
    while(temp!= NULL) //it will keep on executing till it reaches the last node of the
libraryNode list
    {
        c = strcmp(n, temp->song_name); //comparing the input song name with the
song name of the current node
        if(c==0) //if both the song name are same
        {
            for(i=1;i<=k;i++) //loop will run for as many times the user has entered
            {
                f++; //incrementing the counter for loop No.
                printf("\t %d \t\t %s\t\t %s\t\t %ld\n" ,f, temp->song_name, temp-
>artist_name, temp->duration);
            }
            break;
        }
        temp = temp->next2;
    }

    if(c!=0) //if song not found
    {
        printf("-----\n");
        printf("Data not available.\n");
        printf("Please try again \n\n");
    }
}
}

```

```

void shuffle (int l)
{
    int c = 1,a=0,n,j=0,ch,i;
    struct playlistNode*temp;
    temp = start1;
    srand(time(0)); //to generate random numbers every time the code is run
    if(temp == NULL) //to check if the playlistNode list is empty
    {
        printf("playlist is empty.\n");
        printf("Try again please\n\n\n");
    }
}

```

```

else
{
printf("Song Name \t Artist Name\t Duration(in seconds)\n");
for (c = 1; c <=10; c++) //to shuffle 10 times
{
ch = rand() % 1+ 1; //generates random number
temp = start1;
j=0;
while(temp!= NULL)
{
j++; //incrementing the counter
if(j==ch) //if random generated number matches the counter variable
{
printf("%s\t\t %s\t\t %ld\n" , temp->song_name, temp->artist_name, temp-
>duration);
break;
}
temp = temp->next1;
}
}
}
}

```

Screenshots

```
O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O
*
*
*          WELCOME TO          *
*
*          |      I MUSIC      |      *
*
*
*
O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O

** MAIN MENU **
PRESS 1: Go to Music Library
PRESS 2: Go to Playlist
PRESS 3: Exit from App

Enter the choice: 1

          *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 1
Song Name: STAY
Artist Name: Justin Bieber
Duration(in seconds): 195
Insertion of Song Successful

          *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 3
Song Name: Dinero
Artist Name: Trinidad
Duration(in seconds): 214
Insertion of Song Successful

          *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 5
Song Name: Heaven
Artist Name: James Carter
Duration(in seconds): 156
Insertion of Song Successful

          *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 2
Song Name: Woman
Artist Name: Doja_Cat
Duration(in seconds): 425
Insertion of Song Successful

          *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 4
Song Name: Fever
Artist Name: Farruki
Duration(in seconds): 145
Insertion of Song Successful

          *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 6
Song Name: Pepas
Artist Name: Farruko
Duration(in seconds): 425
Insertion of Song Successful
```

```

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 7
Song Name: Freaks
Artist Name: Monet
Duration(in seconds): 356
Insertion of Song Successful

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 2
Sr No.      Song Name      Artist Name      Duration(in seconds)
1           STAY      Justin Bieber    195
2           Woman     Doja_Cat        425
3           Dinero     Trindad         214
4           Fever     Farruki         145
5           Heaven     James_Carter    156
6           Pepas      Farruko         425
7           Freaks      Monet           356
8           Beggin     Maneskin        178

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 3
Enter the Song Name to be searched: Heaven
Sr No.      Song Name      Artist Name      Duration(in seconds)
1           Heaven     James_Carter    156

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 4
Enter the song title to be added to the playlist: STAY
Song Name      Artist Name      Duration(in seconds)
STAY           Justin_Bieber    195
Insertion of song successful

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 1
Sr No.: 8
Song Name: Beggin
Artist Name: Maneskin
Duration(in seconds): 178
Insertion of Song Successful

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 2
Sr No.      Song Name      Artist Name      Duration(in seconds)
1           STAY      Justin Bieber    195
2           Woman     Doja_Cat        425
3           Dinero     Trindad         214
4           Fever     Farruki         145
5           Heaven     James_Carter    156
6           Pepas      Farruko         425
7           Freaks      Monet           356
8           Beggin     Maneskin        178

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 3
Enter the Song Name to be searched: Heaven
Sr No.      Song Name      Artist Name      Duration(in seconds)
1           Heaven     James_Carter    156

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 4
Enter the song title to be added to the playlist: Beggin
Song Name      Artist Name      Duration(in seconds)
Beggin         Maneskin         178
Insertion of song successful

```

```

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 5
Enter the song name to put on loop: Dinero
Number of times you want to loop a song : 4
    Loop No.      Song Name      Artist Name      Duration(in seconds)
    1             Dinero          Trindad          214
    2             Dinero          Trindad          214
    3             Dinero          Trindad          214
    4             Dinero          Trindad          214

                *Menue for MUSIC LIBRARY*
Press 1: Insert songs into the Main Library
Press 2: Display all songs in Main Library
Press 3: Search by Title
Press 4: Insert song in Playlist
Press 5: Enter to put song on Repeat
Press 6: Go to the Playlist Menu
Press 7: Return to Main Menu

Enter choice: 6

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 1
Enter the song title to be added to the playlist: Dinero
    Song Name      Artist Name      Duration(in seconds)
    Dinero          Trindad          214
Insertion of song successful

                **FOR PLAYLIST**

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 1
Enter the song title to be added to the playlist: Damn
    Song Name      Artist Name      Duration(in seconds)
    Damn           Data not available.
Please try again

                **FOR PLAYLIST**

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 1
Enter the song title to be added to the playlist: Pepas
    Song Name      Artist Name      Duration(in seconds)
    Pepas          Farruko          425
Insertion of song successful

                **FOR PLAYLIST**

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 1
Enter the song title to be added to the playlist: Woman
    Song Name      Artist Name      Duration(in seconds)
    Woman          Doja_Cat          425
Insertion of song successful

```



```

**FOR PLAYLIST**

    * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 2
  Sr No.      Song Name      Artist Name      Duration(in seconds)
    1         Woman          Doja_Cat         425
    2         Pepas          Farruko          425
    3         Dinero         Trinidad         214
    4         Beggin         Maneskin         178
    5         STAY           Justin_Bieber    195

**FOR PLAYLIST**

    * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 3
Playlist Duration(in seconds): 1437 sec

**FOR PLAYLIST**

    * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 4
Enter the song name to be searched: STAY
  Sr No.      Song Name      Artist Name      Duration(in seconds)
    1         STAY           Justin_Bieber    195

**FOR PLAYLIST**

    * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 5
Song Name      Artist Name      Duration(in seconds)
Pepas          Farruko          425
Pepas          Farruko          425
Dinero         Trinidad         214
Woman          Doja_Cat         425
Beggin         Maneskin         178
STAY           Justin_Bieber    195
Dinero         Trinidad         214
Woman          Doja_Cat         425
STAY           Justin_Bieber    195
Dinero         Trinidad         214

```

```

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 6

Before Swapping:

    Sr No.      Song Name      Artist Name      Duration(in seconds)
    1           Woman          Doja_Cat         425
    2           Pepas          Farruko          425
    3           Dinero         Trinidad         214
    4           Beggin        Maneskin         178
    5           STAY          Justin_Bieber    195

Enter the song names to be swapped:
Dinero STAY

After Swapping:

    Sr No.      Song Name      Artist Name      Duration(in seconds)
    1           Woman          Doja_Cat         425
    2           Pepas          Farruko          425
    3           STAY          Justin_Bieber    195
    4           Beggin        Maneskin         178
    5           Dinero         Trinidad         214

                **FOR PLAYLIST**

                * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 7
Enter the Sr No. of the song to be delete:
3

Deletion successful
New playlist

    Sr No.      Song Name      Artist Name      Duration(in seconds)
    1           Woman          Doja_Cat         425
    2           Pepas          Farruko          425
    3           Beggin        Maneskin         178
    4           Dinero         Trindad          214

```

```

**FOR PLAYLIST**

      * Menue for PLAYLIST *
Press 1: Insert a song into playlist
Press 2: Display the playlist
Press 3: Display total duration
Press 4: Search by Title
Press 5: Shuffle playlist
Press 6: Swap
Press 7: Delete a song from playlist
Press 8: Return to Music Library
Press 9: Return to Main Menu

Enter choice: 9

      ** MAIN MENU **
PRESS 1: Go to Music Library
PRESS 2: Go to Playlist
PRESS 3: Exit from App

Enter the choice: 3

      O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O
      *
      *          THANK YOU FOR USING OUR APP          *
      *
      O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O*X*O

...Program finished with exit code 0
Press ENTER to exit console.

```

Conclusion

We have successfully used Singly link list as the data structure in the implementation of the music player – I Music code for the execution of various functions and choices for the user to choose from.