|  |
| --- |
| **Data Structures and Algorithms** |
| [Wynk Music App] |
| **Course Project Report** |

|  |
| --- |
| **School of Computer Science and Engineering**  **2022-23** |

**Contents**

|  |  |
| --- | --- |
| **Si. No.** | **Topics** |
| 1. | Course and Team Details |
| 2. | Introduction |
| 3. | Problem Definition |
| 4. | Functionalities |
| 5. | Project Tools |
| 6. | Learning and Takeaway |
| 7. | References |

**1. Course and Team Details**

**1.1 Course details**

|  |  |
| --- | --- |
| **Course Name** | Data Structures and Algorithms  (Theory and Lab) |
| **Course Code** | 22ECAC203 and 22ECAP202 |
| **Semester** | III |
| **Division** | F |
| **Year** | 2022-23 |
| **Mentor** | Rajashekharaiah K M M |

**1.2 Team Details**

|  |  |  |
| --- | --- | --- |
| **Si. No.** | **Roll No.** | **Name** |
| 1. | 151 | Atreya Hampiholi |
| 2. | 153 | Devaj |
| 4. | 155 | Mehwish Nidgundi |
| 3. | 156 | Gouri Vernekar |

**2. Introduction**

Data structures and algorithms provide a foundation for problem-solving in computer science. By understanding different data structures and algorithms, programmers can develop effective solutions to complex problems, regardless of the programming language or platform.

Data structures and algorithms provide efficient ways of storing and manipulating data, which is crucial in software development. Efficient algorithms and data structures can help programs run faster, use less memory, and consume fewer system resources.

As programs grow in size and complexity, data structures and algorithms become increasingly important for ensuring that the program can scale and handle large amounts of data. The ability to manage large amounts of data efficiently is a critical skill in today's world of big data. Data structures and algorithms provide a way to solve a wide variety of problems in different domains, such as web development, artificial intelligence, and robotics. By learning these concepts, programmers can develop a versatile skill set that can be applied in many different contexts.

**3. Problem Statement**

The task is to build a console based music streaming application in which the functions like play specific songs, count the number of songs in the playlist are implemented using Data structures and Algorithms. Wynk is a musical application in which the above-mentioned functions are included.

The drawback of Wynk is that we cannot sort songs in the playlist according to the released year. We had overcome this drawback by implementing an algorithm known as

Bubble sort. Similar to this function many other functions are present in this application which can be built using various algorithms.

**4. Functionalities**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SI. No.** | **Function Name** | **Description** | **DS and Algorithm Used** | **Efficiency** |
|  | insert\_song | Adds a song to the playlist | Circular doubly linked list | O(N) |
|  | delete\_song | Deletes a song from the playlist | Circular doubly linked list | O(N) |
|  | display\_playlist | Prints all the songs present in the playlist | Circular doubly linked list | O(N) |
|  | play\_first\_song | Plays the first song in the playlist | Circular doubly linked list | O(1) |
|  | play\_last\_song | Plays the last song in the playlist | Circular doubly linked list | O(1) |
|  | display\_by\_language | Prints all the songs of a particular language | Circular doubly linked list | O(N) |
|  | play\_prev | Play the previous song | Circular doubly linked list | O(1) |
|  | play\_next | Play the next song | Circular doubly linked list | O(1) |
|  | count | Prints the number of songs present in the playlist | Circular doubly linked list | O(N) |
|  | search\_and\_play | Used to play a song by searching . | Naive’s Algorithm | O(N2) |
| 11. | sort | Sorts all the songs in a playlist by the year released | Bubble sort | O(N2) |
| 12. | display\_recently\_searched | Displays all the recently searched songs | Array of structures | O(N) |
| 13. | display\_curr\_song | Displays currently playing song | Circular doubly linked list | O(1) |

**5. Project Tools**

**5.1 Data Structures and Algorithms**

| **Sl.No.** | **DS and Algorithm Used** | **Description** | **Efficiency** | **Order of growth** |
| --- | --- | --- | --- | --- |
| 1. | Naive’s Algorithm | Naive’s algorithm is simple and easy to implement. It handles both continuous and discrete data. | O(N2) | O(N!) |
| 2. | Circular-Doubly-linked list | Insertion and deletion operations are efficient and easily implemented as compared to other data structures like Array. | O(N) | O(N) |
| 3. | Bubble sort | Bubble Sort is the simplest sorting algorithm that works by repeatedly  swapping the adjacent elements if they are in the wrong order. | O(N2) | O(N2) |
| 4. | Brute Force Algorithm | This is the most basic and simplest type of algorithm. In the end it definitely gives the optimal solution. | —-- | O(N) |
| 5. | Array of Structure | It is used to store the names of recently searched songs and display them when required. | O(N) | O(N) |

**5.2 Project Statistics**

| **Si. No.** | **Measure** | **Value** |
| --- | --- | --- |
|  | Total Functions in Project | 15 |
|  | Total number of lines of code  (Including comments, newlines, etc.) | 1016 |
|  | Number of Errors | 0 |
|  | Number of Warnings | 0 |
|  | Team Satisfaction about Project | 100 |

**6. Learning and Takeaway**

Circular-Doubly-linked list to create a playlist, adding new songs, and removing songs from playlist. From this Data Structure, we learnt how to traverse through circular doubly linked lists and perform the given functions.

Bubble Sort is the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in the wrong order. Bubble sort is easy to understand and implement.

Likewise, we can figure out different solutions for other problems and every approach is explained from different algorithms and data structures.

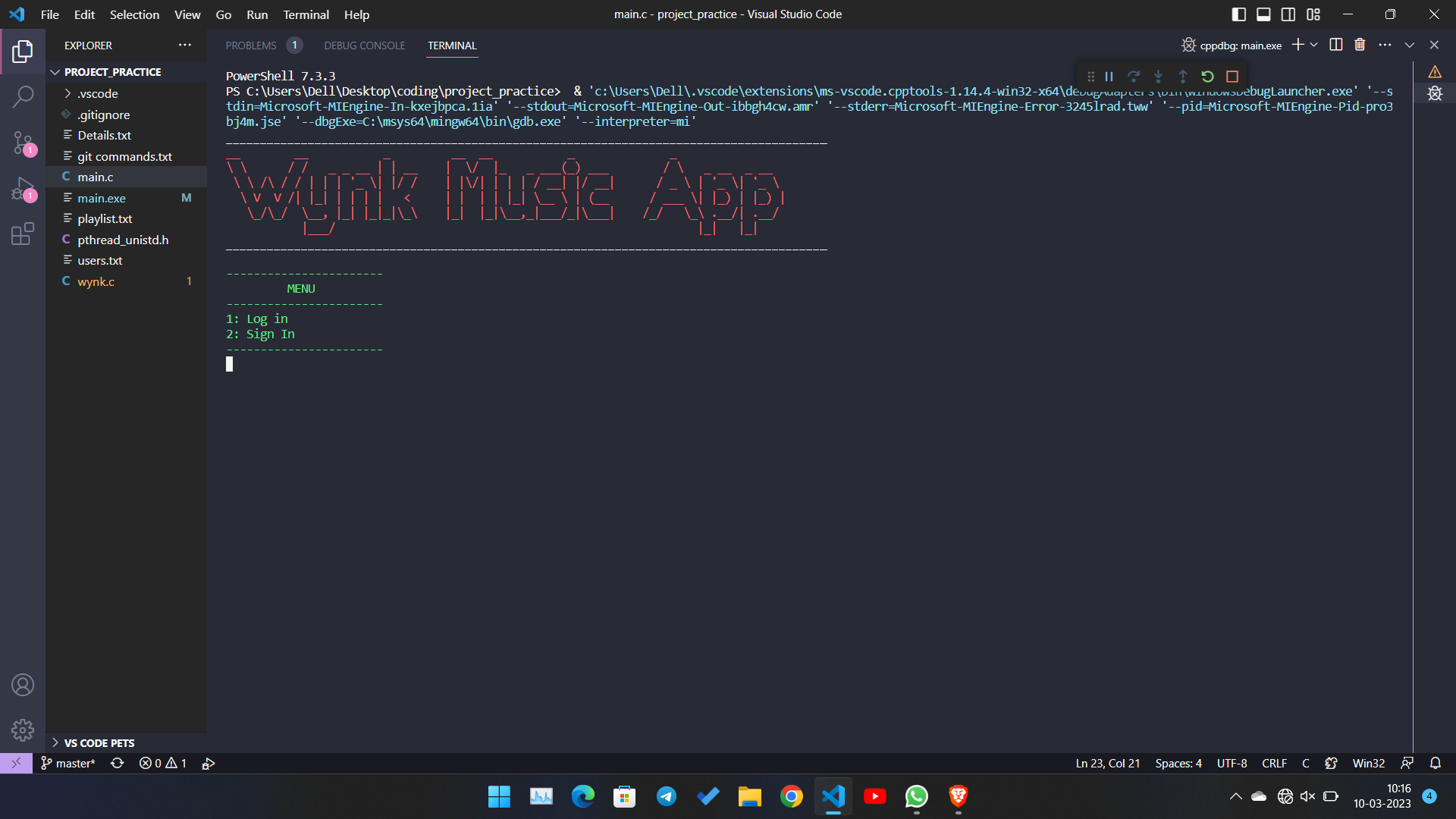
We also learnt how algorithmic techniques are put into solving various real-world problems.

**7. References**

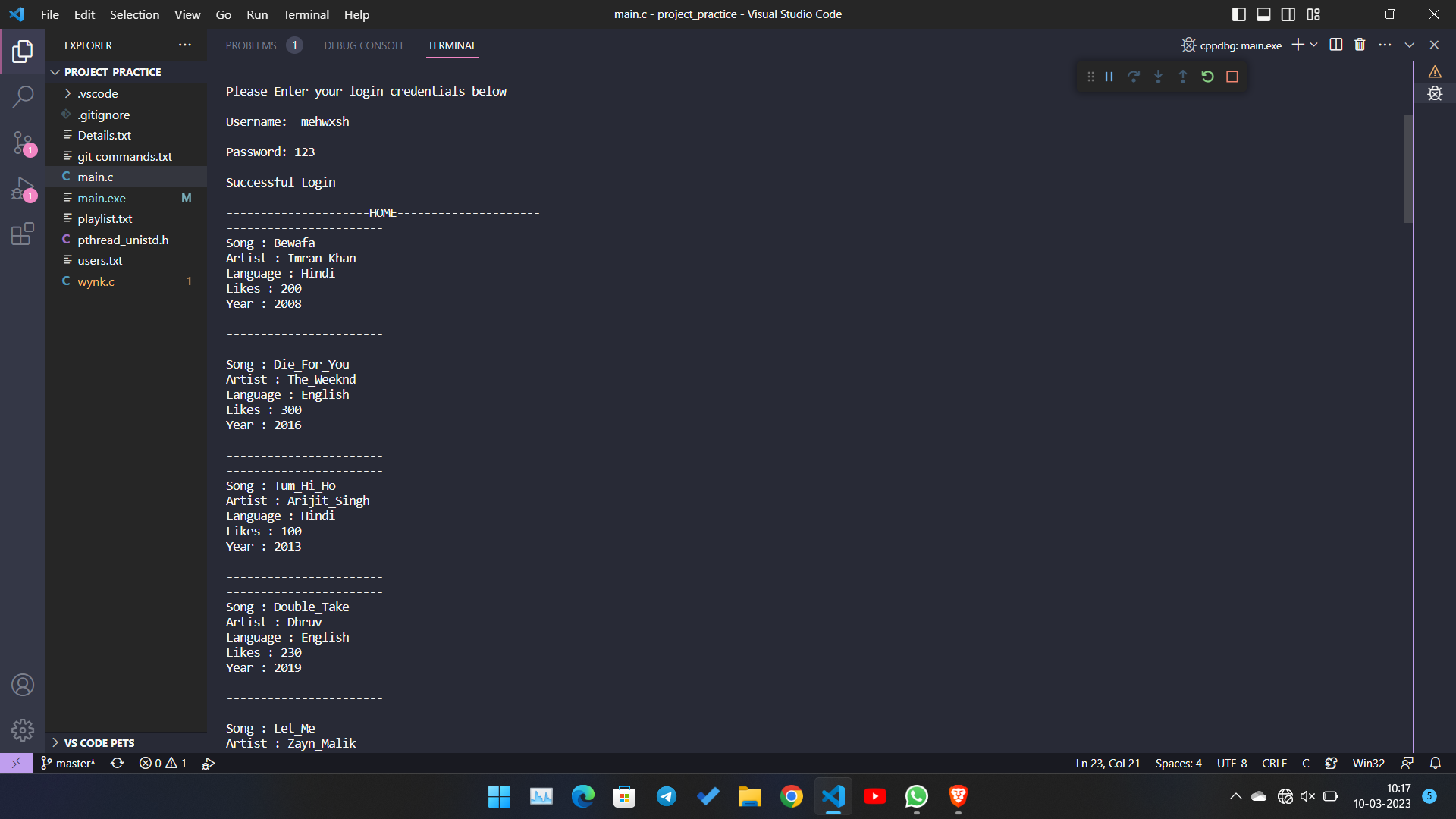
[1] [Data Structures through C by Yashwant Kanetkar](https://www.amazon.in/Data-Structures-Through-fundamentals-through/dp/9355511892?source=ps-sl-shoppingads-lpcontext&ref_=fplfs&psc=1&smid=A1WYWER0W24N8S)

**8. Test cases**

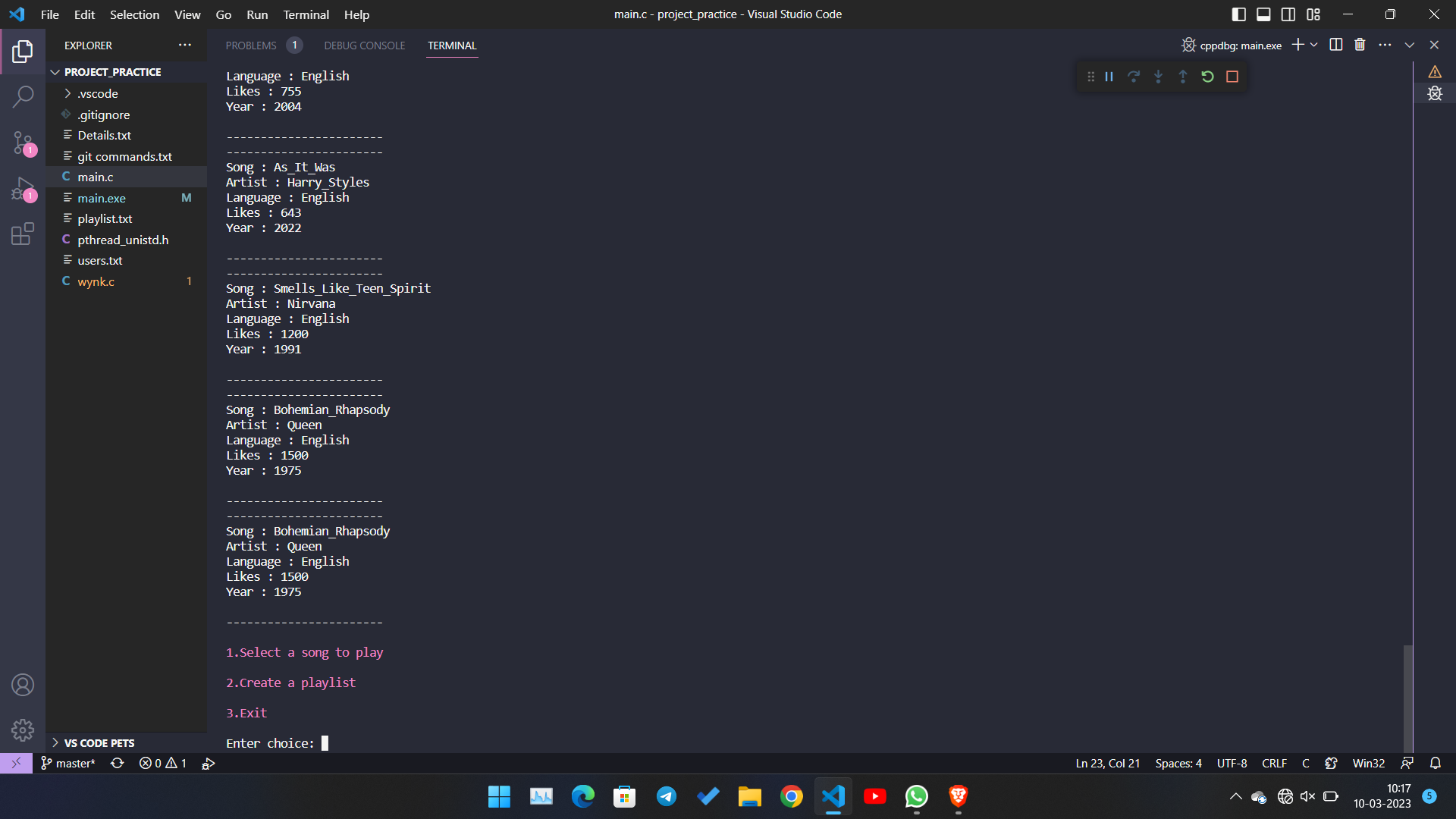
1. **Welcome and login options**

****

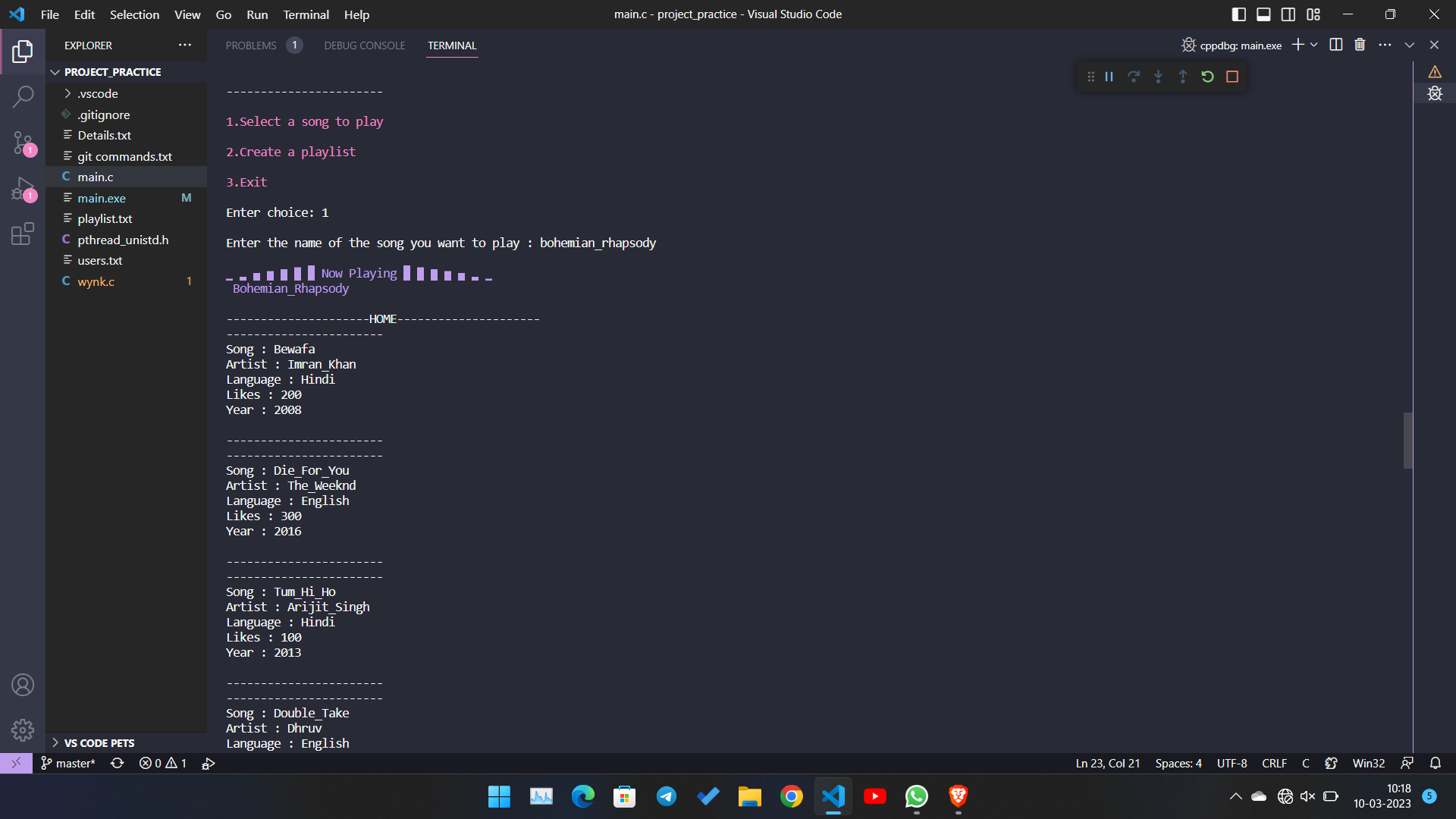
1. **Home console after successful login**

****

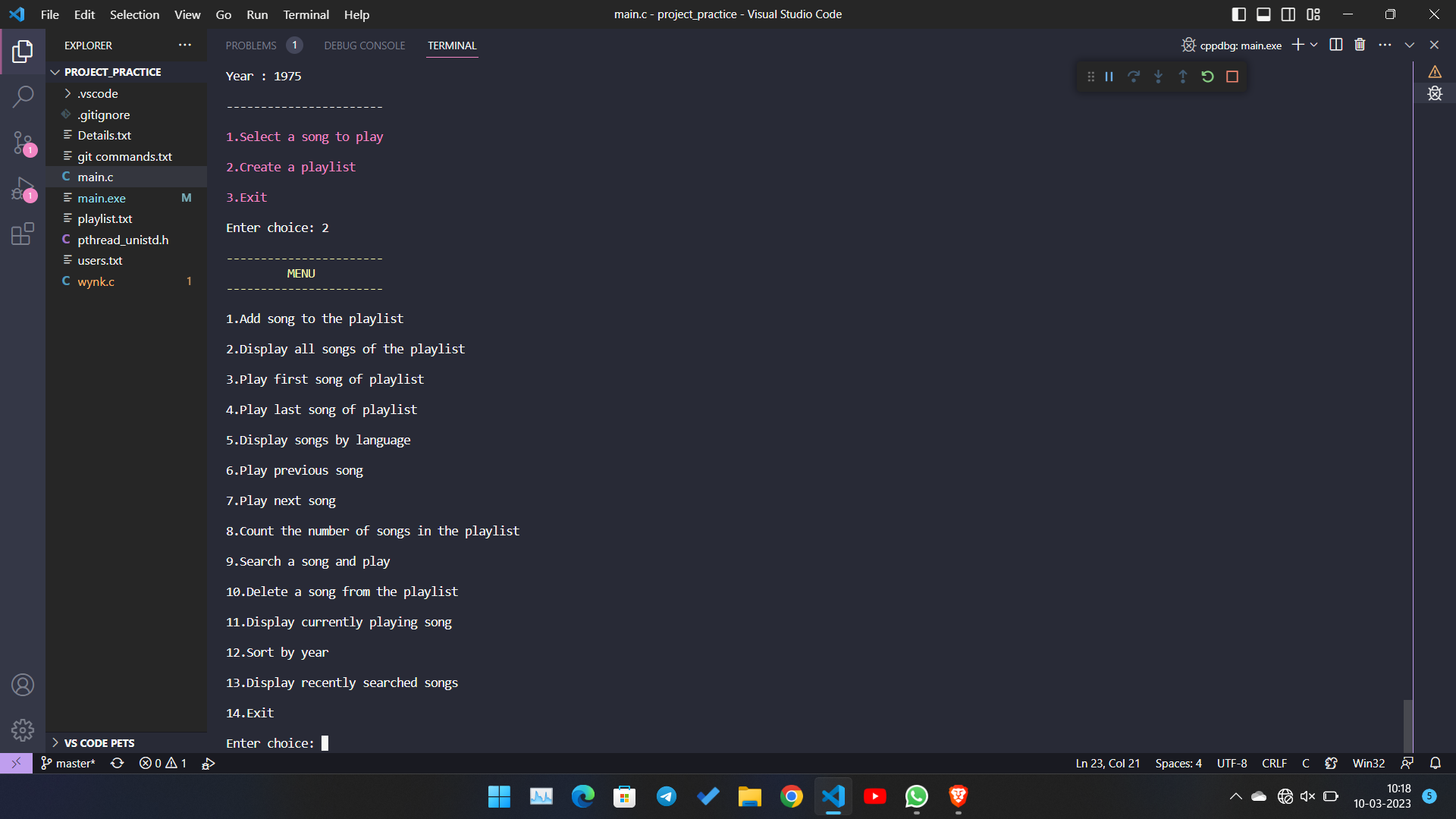
1. **Menu 1**

****

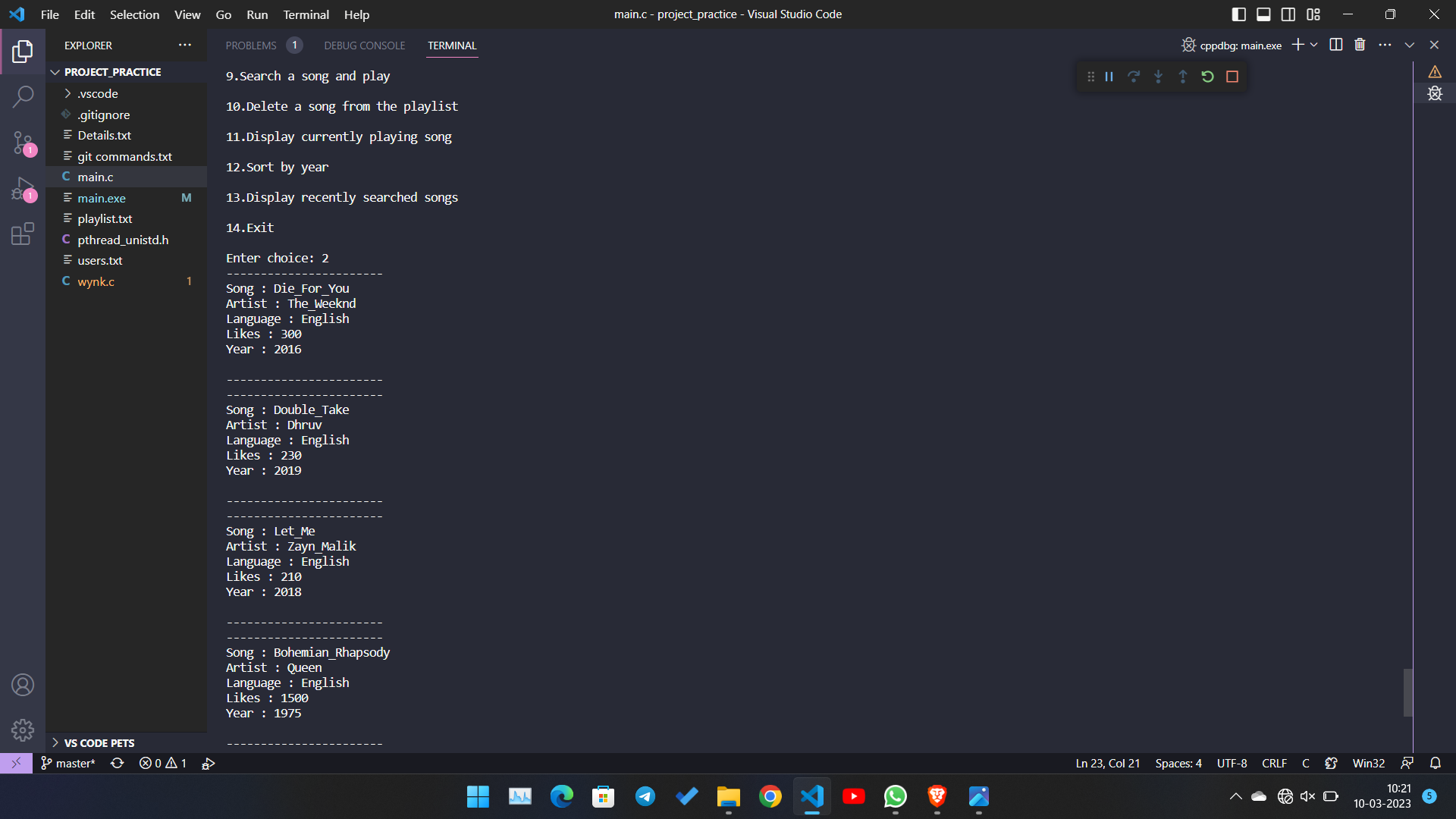
1. **When a song is played from home**

****

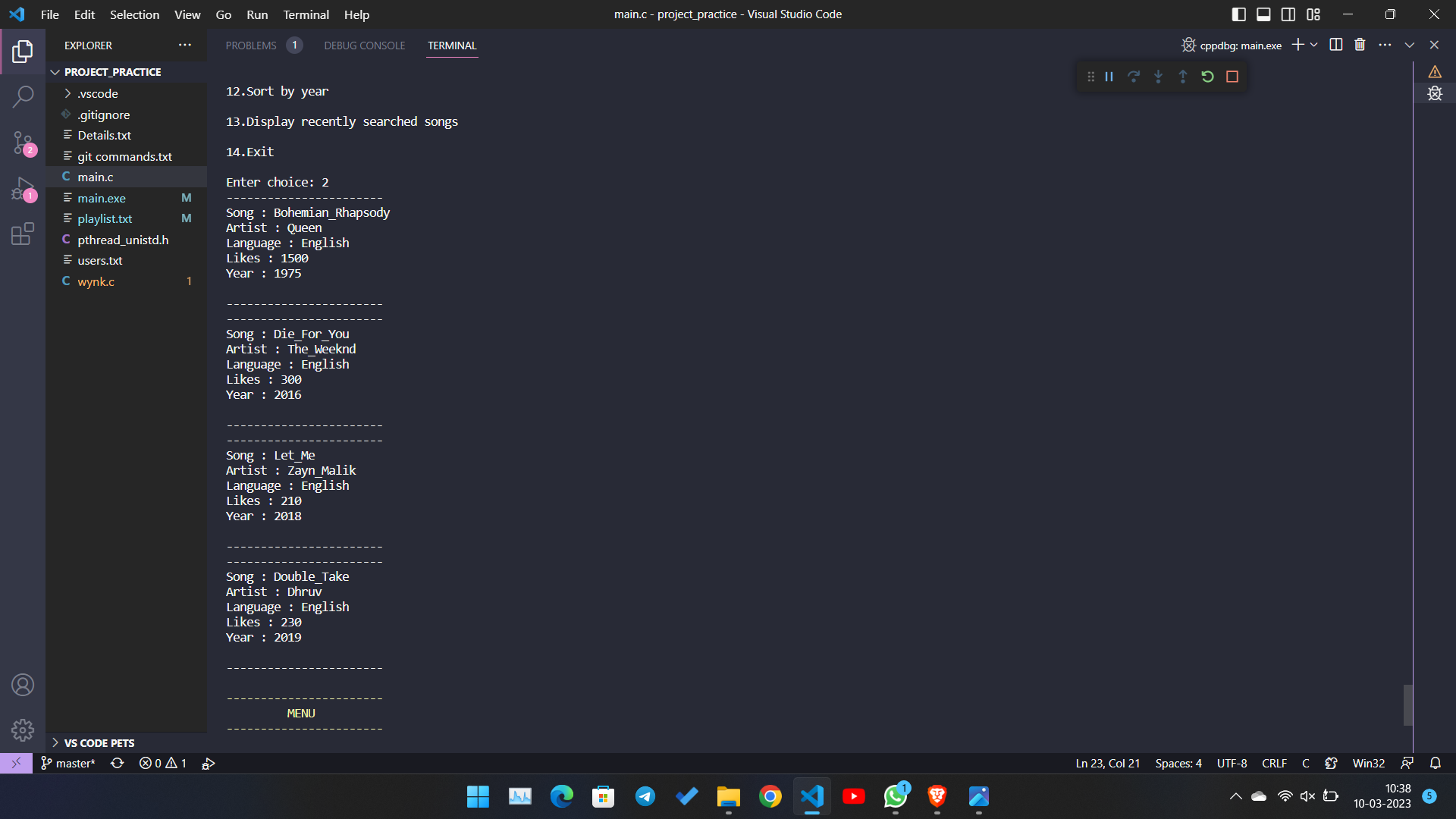
1. **Menu 2**

****

1. **When songs in the playlist are displayed**

****

1. **When songs are sorted by year in the playlist**

****