- **Iced ~~tea~~ Coffee Problem**

Two Peoples meet regularly at a cozy coffee shop to catch up on their lives. However, they tend to interrupt each other while speaking, leading to confusion and frustration. Both of them (Person 1 and Person 2) enjoy iced coffee, but Person 2 frequently misspells it as "iced tea," leading to mix-ups and misunderstandings.

Design a C program that utilizes semaphores, threads, and processes to allow the two friends to communicate effectively and minimize the chances of confusion and mix-ups.

Requirements:

1. The program should simulate a conversation between the two friends, with each speaking in turn.
2. Use semaphores to ensure that only one person speaks at a time, and the other listens. Implement a turn-taking mechanism to ensure that each person has an equal opportunity to speak.
3. Implement error detection and correction for Person 2's tendency to misspell "iced coffee" as "iced tea." Whenever Person 2 mentions "iced tea," the program should automatically correct it to "iced coffee."
4. Use threads or processes to simulate the conversation and error detection/correction mechanisms.
5. The program should output the corrected conversation, including the number of errors detected and corrected.

Your program should demonstrate the efficient and accurate communication between the two friends, minimizing confusion and errors, using semaphores, threads, and processes, while they discuss their mutual love for iced coffee.

- **Acrylic Painting Problem**

Someone loves acrylic painting and is working on a collaborative project with multiple other artists. Each artist is responsible for painting different parts of the artwork, and they must work sequentially to ensure that each part is completed before the next artist can start. To keep track of progress and not lose hope, the artists decide to use a multi-threaded C program with semaphores, processes, threads, and mutex locks to coordinate and monitor the painting process.

The shared artwork has several sections, and each section must be painted in a specific order. The artists also need to wait for the paint to dry before the next artist can begin their part. They require a system to manage this coordination and keep track of progress efficiently.

To address this problem, the artists will create a C program that uses multiple threads to represent the artists working on different parts of the painting. Semaphores will be used to synchronize the order in which the artists work on their sections, allowing time for the paint to dry and ensuring that no two artists work on the same section simultaneously.

Additionally, the artists will use a mutex lock to protect access to a shared progress report that logs the completion of each section. This progress report will help Alice and her fellow artists stay motivated by giving them a clear view of the overall progress of the project. The program will also include processes to handle different stages of the painting, such as preparation, painting, and finishing touches.

By implementing this solution, Alice and her fellow artists will be able to maintain their motivation and complete their collaborative acrylic painting project efficiently and in a well-coordinated manner.

- **Two book enthusiasts**

In a small town, there is a charming and cozy bookshop named “Whispering Pages,” which also serves as a café. Every Saturday, two book enthusiasts meet there to discuss and analyze various books they’ve read, such as “Heart Bones,” “The Alchemist,” and “Kafka on the Shore.” The bookshop frequently hosts multiple book clubs and reading groups that gather simultaneously, leading to lively discussions and debates throughout the shop.

Due to the increasing number of book clubs and reading groups, the bookshop owner realizes that managing the schedules, room allocations, and resources for each group has become quite challenging. The owner wants to implement a multi-threaded C program with semaphores, processes, threads, and mutex locks to efficiently coordinate the activities and ensure a smooth experience for all visitors.

The problem that needs to be addressed involves managing the following aspects:

Scheduling and coordination of book club and reading group meetings, ensuring that no two groups are assigned the same room at the same time.

Resource allocation, including books, tables, chairs, and audio-visual equipment, while avoiding conflicts and double bookings.

Maintaining a shared digital catalog of the bookshop's inventory and keeping it up-to-date, even as multiple visitors and staff access and modify it concurrently.

The C program will need to create multiple threads to represent the different book clubs and reading groups, and use mutex locks to protect access to shared resources such as the room allocation and resource booking systems. Semaphores will be employed to synchronize the activities of the groups and prevent any overlaps in their schedules.

By implementing such a system, the owner of Whispering Pages can ensure that the bookshop continues to operate smoothly and provide a welcoming environment for visitors who come together every Saturday to share their love for literature

- **Book shelf management problem**

Someone living in an apartment decides to create a cozy and well-organized bookshelf in their living room. This person is an avid reader with an extensive book collection and wants to sort and arrange their books efficiently. They decide to use a multi-threaded C program with semaphores, processes, threads, and mutex locks to coordinate the book organization process and ensure the books are correctly sorted and placed on the bookshelf.

The problem that needs to be addressed involves the following aspects:

Sorting the books based on a predefined categorization system, such as alphabetical order, genre, or publication date. This task requires coordinating the different sorting methods and ensuring that the books are sorted correctly and efficiently without disrupting the overall organization of the bookshelf.

Updating a shared digital inventory of the book collection to keep track of the books' locations on the bookshelf. The inventory system should allow the person to easily find and locate any book in the collection.

Ensuring that the digital inventory system is protected from potential data inconsistencies and conflicts, especially if it is accessible from multiple devices or if the person is using an automation system to help with the organization process.

The C program will need to create multiple threads to represent each task involved in the bookshelf organization process. Mutex locks will be employed to protect access to shared resources, such as the digital inventory system, ensuring that it remains consistent and accurate. Semaphores will be used to synchronize the different tasks, allowing them to work together effectively and avoid disrupting the book sorting and organization process.

By implementing such a system, the person can successfully create a cozy and well-organized bookshelf in their living room, providing an enjoyable space for reading and sharing their love for literature.