**1. Introduction to Asynchronous and Multithreaded Programming(Dhirendra Joshi)**

* **Overview of Asynchronous Programming**
  + Importance and benefits
  + Synchronous vs. Asynchronous execution
* **Introduction to Multithreading**
  + Concepts of threads and concurrency
  + Advantages of multithreaded applications

**2. Fundamentals of .NET Task-Based Asynchronous Pattern (TAP)(Pankaj)**

* **Understanding async and await Keywords**
  + Syntax and basic usage
  + Task vs. Task<TResult>
* **Creating and Managing Tasks**
  + Task creation methods
  + Task scheduling and execution

**3. Deep Dive into async and await(Balaji)**

* **Asynchronous Methods**
  + Defining and calling asynchronous methods
  + Return types for asynchronous methods
* **Exception Handling in Asynchronous Code**
  + Try-catch with async methods
  + Task.Exception property
* **Configuring await Behavior**
  + ConfigureAwait method
  + Context capturing and synchronization contexts

**4. Multithreading in .NET(Satya)**

* **Thread Class and Basic Thread Operations**
  + Creating and starting threads
  + Thread lifecycle and states
* **Thread Synchronization**
  + Locks (lock statement)
  + Monitor class
  + Mutexes, Semaphores, and Readers/Writers locks

**5. The Task Parallel Library (TPL)(Satya)**

* **Introduction to TPL**
  + Benefits over traditional threading
  + Core components and architecture
* **Parallel Programming Constructs**
  + Parallel.For and Parallel.ForEach
  + PLINQ (Parallel LINQ) for data processing

**6. Asynchronous Streams and Data Flow(Satya)**

* **Async Streams (IAsyncEnumerable<T>)**
  + Creating and consuming async streams
  + Advantages in data processing
* **Dataflow Library**
  + Building data processing pipelines
  + Blocks and their interactions

**7. Advanced Multithreading Concepts(Satya)**

* **Thread Pool Management**
  + Understanding the .NET Thread Pool
  + Configuring thread pool settings
* **Cancellation and Timeout**
  + Using CancellationToken and CancellationTokenSource
  + Implementing timeouts in asynchronous operations
* **Concurrency Patterns**
  + Producer-Consumer
  + Reader-Writer
  + Immutable Data Structures

**8. Synchronization Primitives and Concurrent Collections(Shiva)**

* **Synchronization Primitives**
  + AutoResetEvent and ManualResetEvent
  + Barrier and CountdownEvent
* **Concurrent Collections**
  + ConcurrentDictionary<TKey, TValue>
  + ConcurrentBag<T>, ConcurrentQueue<T>, and ConcurrentStack<T>
  + Usage scenarios and performance considerations

**9. Best Practices for Asynchronous and Multithreaded Programming(Prasanna)**

* **Avoiding Common Pitfalls**
  + Deadlocks and race conditions
  + Thread starvation and context switching overhead
* **Optimizing Performance**
  + Minimizing synchronization overhead
  + Efficient task management
* **Design Patterns for Concurrency**
  + Producer-Consumer Pattern
  + Task-based Asynchronous Pattern (TAP)
  + Async/Await Best Practices

**10. Debugging and Testing Asynchronous and Multithreaded Code(Prasanna)**

* **Debugging Tools and Techniques**
  + Visual Studio debugging for async and multithreaded code
  + Analyzing thread dumps and call stacks
* **Unit Testing Asynchronous Code**
  + Writing effective async unit tests
  + Using testing frameworks (e.g., xUnit, NUnit) with async code
  + ========================