**Programming 2B**

**PROG6212**

**ICE Task**

**How LINQ will be used for filtering and organizing data**

LINQ will be employed to filter and categorize events based on criteria like Date, Type, or Department. Using LINQ's querying capabilities, you can:

* Retrieve events scheduled for a specific date.
* Filter events by department or type (e.g., academic, sports, cultural).
* Sort events chronologically or alphabetically by title.

Anonymous Types:

* You can use anonymous types to select only specific details of events when displaying them, such as Title and Date, without needing a dedicated class for every possible data combination.

**Integration of asynchronous operations and multithreading**

* Fetching Event Data from External Sources: Use async methods to fetch and process event data from external sources or APIs without blocking the main thread, improving the system's responsiveness.
* Integrating with Calendar Services: Asynchronous operations can be used to integrate event details with external calendar services (e.g., Google Calendar), allowing users to synchronize their schedules.

**Concurrency with Multithreading**

* Notifying Students: You can use multithreading to send notifications concurrently to multiple students, ensuring that all receive timely updates without delays.
* Updating Event Status: Concurrent processing will allow real-time updates to event statuses (e.g., from "Scheduled" to "Ongoing" or "Completed").
* Managing Registrations: Multithreading ensures efficient handling of multiple registration requests simultaneously, preventing bottlenecks during peak registration times.