FOOD PROCESSING PLANT

Presented By Group-4

Team members

BHARATH KUMAR --- CB.EN.U4CSE22507

SUJAN KARTHIK --- CB.EN.U4CSE22547

N SIDDHARTH --- CB.EN.U4CSE22529

NANDHAKUMAR --- CB.EN.U4CSE22530

KAMALAKRISHAN --- CB.EN.U4CSE22519

Chapter 2

Scenario: Food Processing Plant

Imagine a food processing plant where vegetables go through various stages such as washing, peeling, chopping, and packaging. These stages share resources like conveyor belts, washing stations, storage bins, and packing machines. The challenge is to allow multiple stages to read or access shared resources, while ensuring that only one stage writes or modifies the resource at a time to maintain data integrity and avoid bottlenecks.

Chapter 2

Reader-Writer Problem in a Food Processing Plant

In a food processing plant focused on vegetable processing, various stages like washing, peeling, chopping, and packaging happen in sequence. Let's consider a scenario where some stages involve reading from a shared resource (like conveyor belts carrying vegetables), while other stages involve writing to the resource (like adding or removing items from the conveyor belt).

Mutual Exclusion

Chapter 2

- **Concept**: Ensures that only one writer at a time can modify a shared resource, preventing conflicts or corruption.
- **Application**:
  - Consider a food processing plant where different stages are involved in vegetable processing, such as washing, peeling, chopping, and packaging. These stages might share common resources like conveyor belts, storage bins, or processing machines. Mutual exclusion ensures that only one stage can access a shared resource at a time, preventing overlapping operations that could lead to errors or safety risks.
- Solution
  - Locks on Conveyor Belts: Implement locks on conveyor belts. Before a stage can place or remove vegetables from the conveyor belt, it must acquire a lock, ensuring mutual exclusion.
  - Example: If the chopping stage is using the conveyor belt to move chopped vegetables to the packaging area, the peeling stage must wait until the chopping is complete to avoid interference.

Chapter 2

Progress

- **Concept**: : Guarantees that operations continue without indefinite blocking, allowing each stage to progress.
- **Application**:
  - Suppose the food processing plant has a conveyor belt system used by different stages. If there is no clear system to determine which stage accesses the conveyor belt, it could lead to a situation where multiple stages wait for others to finish, causing a deadlock. For example: The peeling stage waits for the washing stage to finish, but the washing stage is waiting for the chopping stage, creating a circular dependency that prevents progress.
- Solution
  - This can be solved by Designing the workflow such that stages acquire resources in a fixed order, reducing the chances of deadlock.

Chapter 2

Bounded Waiting

- **Concept**: : Limits the time a process can wait before gaining access, avoiding starvation.
- **Application**:
  - In a food processing plant focusing on vegetable processing, various stages like washing, peeling, chopping, and packaging occur in a sequence. The shared resource in this context might be conveyor belts, processing machines, or storage containers.
  - The bounded waiting problem arises when a process has to wait indefinitely because of a scheduling or resource access issue.
- Solution
  - **Time Slots**: Assign a maximum time slot for each stage to use the shared resource (conveyor belt). This ensures that no stage can monopolize the resource for too long, allowing others to proceed.

THANK YOU