Computer Simulation Assignment

Group Members:

1. Mohammad Abubakar Atiq (F2022031002)
2. Barira Qasim (F2022031016)
3. Hannan Ahmed Siddiqui (F2022031010)

Topic: Simulation of Student Flow Through University Entrance and Exit Gates

---

1. Problem Formulation

The goal of this simulation is to model and analyze student flow through the University of Management and Technology's entry and exit gates. The primary objectives are:

- To observe student entry/exit flow rates at various times, especially during peak hours.
- To identify and alleviate bottlenecks, particularly at the emergency gate.
- To simulate different scenarios that may optimize gate efficiency and reduce congestion.

Problem Statement:
The university has two main entrance and one exit gate for girls, two entrance and two exit gates for boys, and one emergency gate, which uses an RFID system for student ID validation. It has been observed that, on average, 10 students enter and 3 students exit per minute through the emergency gate. Additionally, student flow rates reduce between 3:30 pm and 5:00 pm compared to the earlier slot, as reported by the guards.

---

2. Data Collection

We collected data based on observations and consultations with university security. The data focuses on the following:

1. **Student Flow Rates**:
    - **Emergency Gate**: 10 students enter and 3 exit per minute on average.
    - **Other Gates**: Flow rates at other gates are assumed to align with typical entry/exit rates at the university during peak times (to be estimated or observed directly).
2. **Time Slot Analysis**:
    - Between 3:30 pm and 5:00 pm, student flow significantly decreases, indicating a lower demand for gate access.
3. **Entry/Exit Distribution**:
    - We assume a balanced distribution of students using different gates based on proximity and availability, with a slight preference for the emergency gate during peak hours.

---

3. Model Creation

The model simulates the student flow at each gate, accounting for time-based variations. Key components of the model include:

- **Entities**: Students entering or exiting through designated gates.
- **Attributes**: Attributes include student ID (validated via RFID), gender, and gate choice.
- **Queues**: Lines forming at each gate, particularly during peak hours.
- **Processes**:
    - Entry and exit verification through the RFID system at each gate.
    - Distinction between peak and off-peak hours to simulate flow changes.
- **Resources**: Physical gates and RFID validation stations.

---

4. Translate the Model

Using Arena or any suitable simulation software, we translated our model into the following setup:

1. **Entities and Resources**: Each gate is treated as a resource with defined capacities.
2. **Processes**: The flow includes RFID validation for entry and exit processes.
3. **Time-Dependent Variables**: We introduced time-based conditions, where between 3:30 pm and 5:00 pm, entry and exit rates are reduced to reflect lower student flow.
4. **Logic Flow**: A block diagram or flowchart is used to visualize the entry, queue, RFID validation, and exit sequences.

Diagram: A block diagram showing the student flow through each gate, with differentiated entry and exit processes for boys, girls, and emergency gate usage, plus a time-based condition for peak and off-peak periods.

---

5. Verification and Validation

1. **Verification**: We tested the model with a small sample size to ensure that entry/exit processes and flow logic aligned with expected behavior.
2. **Validation**: The model was validated by comparing its results against observed flow rates, especially during peak hours and off-peak hours from 3:30 pm to 5:00 pm. Adjustments were made based on any discrepancies between the simulated and observed data.

---

6. Results and Analysis

By running the simulation under various scenarios (peak vs. off-peak hours):

- We identified potential congestion points at the emergency gate during peak hours.

- We observed that, as expected, flow decreased significantly between 3:30 pm and 5:00 pm, indicating a natural alleviation of congestion.

Potential Recommendations:

- Increase RFID validation speed at the emergency gate or encourage alternative gate usage during peak hours.
- Consider adjustments to gate operations during specific times based on student flow patterns.

This simulation provided insights into the efficiency of the university's entry and exit gates and identified opportunities for managing student flow more effectively.

ARENA PROCEDURE

Step-by-Step Procedure to Create the Simulation Model in Arena

1. Open Arena Software

- Launch Arena simulation software on your computer.

2. Data Definitions Setup

- Go to the **Data** panel to set up the necessary data, such as Resources, Entities, and any Variables you may need to track within the model.

A. Resource Definitions:

- Define resources representing each gate.
    - **Name**:
        - Girls Entrance Gate
        - Boys Entrance Gate
        - Emergency Girls Gate
        - Emergency Boys Gate
    - **Type**: Select **Fixed Capacity**.
    - **Capacity**: Set to **1** (assuming one student passes through at a time).
    - **Failures**: Set to **0 rows** if no expected downtime is planned.

B. Entity Definitions:

- **Entity Name**: Student
- **Entity Type**: Set the student ID as **String** to manage unique entries and tracking.

3. Create Modules for Student Arrivals

- Use **Create** modules to represent students arriving at each gate. Separate modules will help manage specific entry points and arrival patterns.

A. Emergency Girls Gate Create Module - **Name**: Emergency Girls Arrival - **Entity Type**: Student - **Type**: Set to **Random (Expo)** for arrivals with a random pattern. - **Value**: Set the arrival rate, such as **10 students per minute**. - **Units**: Set to **Minutes**. - **Entities per Arrival**: Set to **1**. - **Max Arrivals**: Leave blank or set to **Infinite**. - **First Creation**: Set to **0**.

B. Emergency Boys Gate Create Module - **Name**: Emergency Boys Arrival - **Repeat similar settings as above**, adjusting for specific arrival rates if needed.

C. Girls Entrance Create Module - **Name**: Girls Entrance Arrival - **Adjust similar settings**, setting an appropriate arrival rate.

D. Boys Entrance Create Module - **Name**: Boys Entrance Arrival - **Adjust similar settings** as above.

4. Process Modules for RFID Validation

- Drag and drop **Process** modules for each gate to simulate the RFID validation process.

A. RFID Validation Girls Entrance - **Name**: RFID Validation Girls Entrance - **Type**: **Standard** - **Logic**: Set to **Seize Delay Release** - **Priority**: Medium priority (**2**) - **Resources**: Select **Girls Entrance Gate** - **Delay Type**: **Triangular** to reflect variability in process time. - **Units**: **Minutes** - **Allocation**: **Value Added** - **Minimum**: Set to **0.05 minutes** - **Value**: Set to **0.1 minutes** - **Maximum**: Set to **0.2 minutes**

B. RFID Validation Boys Entrance - Repeat the above steps, naming this module RFID Validation Boys Entrance and linking it to **Boys Entrance Gate**.

C. RFID Validation Emergency Girls - Repeat the steps, connecting it to **Emergency Girls Gate**.

D. RFID Validation Emergency Boys - Repeat the steps, connecting it to **Emergency Boys Gate**.

5. Add Discrete Processing Modules for Exit Handling (If Needed)

- To model students leaving the gates, you may use the following modules:

A. Dispose Module: - Place this module for students exiting the simulation once their entry process completes.

B. Delay Module: - Use this to represent any exit-related delays, if applicable.

C. Gather Module: - Place a **Gather** module to consolidate students from different paths before exiting, if you need to track cumulative exit rates or process flows.

6. Additional Discrete Modules for Flow Control (Optional)

- **Assign** or **Assign Attribute**: Use to set or adjust any specific attributes like Arrival Time for tracking purposes.
- **Clone**: Use if you need multiple identical student entities.
- **Station and Route**: Use **Station** and **Route** for routing students across different process points, if you have stations defined within the model.

7. Connect the Modules

- Connect the modules in a sequence based on the student journey:
    - **Create → Process (RFID Validation) → Dispose** (or other specified exit modules).
- Ensure logical flow to represent the real-life movement and processes students experience.

8. Run the Simulation

- After ensuring all components are correctly linked and setup is complete, click on the **Run** button to simulate the model.
- Review the output reports to analyze gate usage, student throughput, and any bottlenecks in the system.

9. Verification and Validation

- Verify and validate by comparing the simulation data with actual observations and guard records.
- Adjust parameters to better align with real-world data if discrepancies arise.

Flowchart Structure for Gate Entry Simulation Model

1. **Start (Oval)**
    - Label this as the **Start** of the simulation process.
2. **Arrival Process for Students (Parallelogram)**
    - **Create Modules**:
        - **Emergency Girls Arrival**
        - **Emergency Boys Arrival**
        - **Girls Entrance Arrival**
        - **Boys Entrance Arrival**
    - Each **Create** module should have a label describing which gate the students are arriving at and their rate of arrival.
3. **Decision for Gate Selection (Diamond)**
    - Add a **Decision node** here to represent which gate students will select.
        - Label as "Gate Selected" with paths leading to the respective validation process for each gate:
            - Girls Entrance
            - Boys Entrance
            - Emergency Girls Gate
            - Emergency Boys Gate
4. **RFID Validation Process (Rectangle)**
    - **Process Modules** for RFID validation at each gate:
        - **Girls Entrance RFID Validation**
        - **Boys Entrance RFID Validation**
        - **Emergency Girls Gate RFID Validation**
        - **Emergency Boys Gate RFID Validation**
    - Each process should show the specific **Seize Delay Release** and **triangular distribution** setup.
5. **Exit Decision (Diamond)**
    - Add a **Decision node** if there are conditional exits or a process that students must complete after validation.
    - For example:
        - "Proceed to Exit?" with **Yes** leading to **Exit Module** and **No** possibly leading to a delay or return to entry point.

6. **Dispose Module (Parallelogram)**
    - This module represents students exiting the simulation after completing the gate entry and validation.
    - Label this as **Student Exit**.
7. **End (Oval)**
    - Label the **End** point of the simulation process.