

Lab 01
CSE 156
Introduction to Programming Lab
Fall 2025

1. Write a Java program that takes a student's score (0-100) as input and categorizes it as follows:

- 90-100 → "Excellent"
- 75-89 → "Good"
- 50-74 → "Average"
- Below 50 → "Fail"

2. Write a program that takes three integers as input and determines the largest among them using nested `if-else`.

3. Write a Java program to check if a given year is a leap year using the following conditions:

- A year is a leap year if it is divisible by 400.
- If not, it is a leap year if it is divisible by 4 but **not** by 100.

4. Write a Java program that takes a single character as input and determines if it is:

- a **vowel** (a, e, i, o, u)
- a **consonant**
- a **digit** (0-9)
- a **special character**

5. Write a Java program that displays a menu to the user:

1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit

6. Write a Java program to calculate the electricity bill based on the number of units consumed using the following rate chart:

- Up to 100 units: TK 2 per unit
- 101-300 units: TK 3 per unit
- 301-500 units: TK 5 per unit
- Above 500 units: TK 7 per unit

7. Write a program that calculates an employee's bonus based on years of service and salary:

- If **years of service** is greater than 5,
 - If salary is **less than TK 50,000**, bonus = 10% of salary
 - Otherwise, bonus = 5% of salary
- If years of service is **less than or equal to 5**, bonus = 2% of salary

8. Write a program that takes three angles of a triangle as input and determines if the triangle is:

- **Equilateral** (all sides equal)
- **Isosceles** (two sides equal)
- **Scalene** (no sides equal)

9. A store gives discounts based on the total purchase amount:

- If the total purchase is **BDT 5,000 or more**,
 - If the customer is a **loyal member**, discount = 20%
 - Else, discount = 10%
- If the total purchase is **less than BDT5,000**, discount = 5%

Calculate and display the final amount after applying the discount.

Story Based Problem

1) Course Seating Overflow

Story: You walk into a packed intro class and there are students hovering by the door. The TA is begging for order. The lab room has stools, the lecture hall does not. You decide who sits, who waits, and who bails.

Inputs:

enrolled, seats, waitlist_count, is_lab_room (true or false)

Rules:

- If enrolled \leq seats, status = Seat Assigned.
- Else if not is_lab_room and waitlist_count < 5 , status = Add to Waitlist.
- Else if is_lab_room and waitlist_count == 0, status = Open Temporary Bench.
- Else status = Section Closed.

Output: status.

2) Restaurant Table Assignment

Story: Dinner rush. A VIP family walks in right after a couple with a reservation. Only a few tables left. You are the gatekeeper of hunger and happiness.

Inputs:

group_size, has_reservation, vip (true or false), free_tables_small, free_tables_large

Rules:

- If vip and free_tables_large > 0, assign Large VIP table.
- Else if has_reservation and group_size <= 4 and free_tables_small > 0, assign Small Reserved table.
- Else if group_size > 4 and free_tables_large > 0, assign Large table.
- Else if group_size <= 4 and free_tables_small > 0, assign Small table.
- Else assign Waiting list.

Output: assigned_table.

3) Cloud VM Sizing Quick Pick

Story: Your startup launches a demo in 2 hours. The CTO wants burst capacity, finance wants a bill that does not cause tears. You pick.

Inputs:

vcpu_needed, ram_gb_needed, burst_ok (true or false)

Rules:

- If vcpu_needed <= 2 and ram_gb_needed <= 4 and burst_ok, plan = t class.
- Else if vcpu_needed <= 4 and ram_gb_needed <= 16, plan = m class.
- Else if vcpu_needed <= 8 and ram_gb_needed <= 32, plan = c class.
- Else plan = custom contact sales.

Output: plan.

4) Metro Fare With Peak

Story: Peak hour chaos, a student card, and the train doors are beeping. Count fast. Decide faster.

Inputs:

distance_km, is_peak (true or false), has_student_card (true or false)

Rules:

- Base fare = 30 if distance_km <= 5, 45 if <= 12, 60 otherwise.
- If is_peak, add 10.
- If has_student_card, subtract 15 but never below 20.

Output: final fare integer and short note.

5) Quick Loan Approval Gate

Story: A borrower arrives with glossy paperwork and confidence. You have 60 seconds to green light, push to review, or hard pass.

Inputs:

credit_score, income, has_collateral (true or false), active_loans

Rules:

- If credit_score >= 780 and income >= 100000 and active_loans == 0, result = Instant Approve.
- Else if credit_score >= 720 and income >= 70000, result = Manual Review.
- Else if

has_collateral and income >= 50000, result = Secured Review.

- Else result = Reject.

Output: result.

6) Grading With Grace

Story: The exam was brutal and half the class got sick. One student plagiarized. You need fair and firm.

Inputs:

raw_score (0 to 100), had_medical_proof (true or false), plagiarism_flag (true or false)

Rules:

- If plagiarism_flag, grade = F, reason = Academic misconduct.
- Else if raw_score >= 90, grade = A.
- Else if raw_score >= 80, grade = B.
- Else if raw_score >= 70, grade = C.
- Else if raw_score >= 60, grade = D.
- Else if had_medical_proof and raw_score >= 50, grade = D (grace).
- Else grade = F.

Output: grade and reason when applicable.

7) Parking Gate Decision

Story: A siren approaches while a sedan with low balance rolls up. The attendant looks at you.

Inputs:

tag_valid (true or false), balance, has_ticket (true or false), is_emergency_vehicle (true or false)

Rules:

- If is_emergency_vehicle, action = Open gate now.
- Else if tag_valid and balance >= 100, action = Open with tag.
- Else if has_ticket, action = Open with ticket.
- Else action = Deny and alert.

Output: action.

8) Minor Bug Triage

Story: A few users report a quirky glitch. Nothing is on fire. Engineers want a sane priority.

Inputs:

severity (Low or Medium), customer_count_affected, sla_hours_remaining

Rules:

- If severity == Medium and customer_count_affected >= 50, priority = P2.
- Else if severity == Medium, priority = P3.
- Else if severity == Low and sla_hours_remaining < 12, priority = P3.
- Else priority = P4.

Output: priority.

9) Simple Content Filter

Story: The forum is lively. A user swore and a staff post leaked personal info. Feelings are not a policy, rules are.

Inputs:

contains_profanity (true or false), contains_pii (true or false), source (staff or user)

Rules:

- If it contains_pii, action = Block and mask.
- Else if contains_profanity and source == user, action = Warn and allow.
- Else if contains_profanity and source == staff, action = Escalate to HR.
- Else action = Allow.

Output: action.

10) Sports Ticket Tier

Story: It is game day. Opponent rank is high, crowd is loud, and pricing needs to match demand without a spreadsheet meltdown.

Inputs:

opponent_rank, home_game (true or false), day_of_week (Fri, Sat, Sun, Mon, ...)

Rules:

- If opponent_rank <= 5 and home_game, tier = Premium.
- Else if day_of_week is Fri or Sat, tier = High.
- Else if opponent_rank <= 15, tier = Medium.
- Else tier = Value.

Output: tier.