INDIAN INSTITUTE OF TECHNOLOGY MADRAS ZANZIBAR

PROJECT GRADING RUBRIC

Z5007: Programming and Data Structures

M.Tech Data Science & AI

Total Project Score: 100 Points (Scales to 25% of Course Grade)

Overview

This rubric provides detailed criteria for evaluating all three project milestones. Instructors will use this to assign points. Students should review this before starting work on each milestone.

MILESTONE 1: Project Proposal (10 Points)

Due: November 21, 2025 (Week 7)

Criterion	Points	Description
A. Content Completeness (4 points)		
Excellent	4	All required sections present and thorough. Clear project option, detailed technical approach, specific dataset identified, realistic timeline, thoughtful challenges identified.
Good	3	Most sections present. Some details missing. Project feasible but needs more specificity.
Satisfactory	2	Basic sections present but lacking detail. Approach unclear or dataset not specific.
Poor	0-1	Missing major sections. Vague or unrealistic plan.
B. Technical Feasibility (3 points)		
Excellent	3	Project scope appropriate for 9 weeks. Data structures well-chosen and justified. Algorithm selection appropriate. Clear understanding of complexity.
Good	2	Project feasible but may be slightly too ambitious or too simple. Data structures reasonable. Some complexity understanding shown.
Satisfactory	1	Project may not be feasible in timeframe. Data structure choices questionable. Limited complexity understanding.
Poor	0	Project clearly not feasible. Inappropriate data structures. No complexity analysis.
C. Appropriate Scope (2 points)		
Excellent	2	Project is neither too ambitious nor too simple. Clear deliverables. Realistic for team size and timeframe.
Good	1.5	Scope mostly appropriate but may need minor adjustments.
Satisfactory	1	Scope needs significant adjustment (too large or too small).
Poor	0	Scope completely inappropriate.
D. Presentation Quality (1 point)		

Criterion	Points	Description
Excellent	1	Professional formatting. Uses provided template. Clear writing. No grammatical errors. Proper citations.
Good	0.75	Good formatting. Minor errors.
Satisfactory	0.5	Acceptable but informal. Several errors.
Poor	0	Poor formatting. Many errors. Hard to read.

MILESTONE 2: Progress Report (30 Points)

Due: December 26, 2025 (Week 11)

Part A: Code Implementation (20 points)

Criterion	Points	Description
1. Core Data Structures (8 points)		
Excellent (7-8)	7-8	At least 2 major data structures fully implemented with all key operations working. Proper class structure with initialization, insertion, deletion, search. Edge cases handled.
Good (5-6)	5-6	2 data structures implemented but some operations incomplete or buggy. Basic functionality works.
Satisfactory (3-4)	3-4	1 data structure complete or 2 partially done. Many operations missing or broken.
Poor (0-2)	0-2	Minimal data structure implementation. Most operations not working.
2. Basic Functionality (8 points)		
Excellent (7-8)	7-8	Can load data successfully. Core algorithm partially working. Produces meaningful output. Main workflow functional.
Good (5-6)	5-6	Can load data. Algorithm started but not fully working. Some output produced.
Satisfactory (3-4)	3-4	Data loading works but algorithm barely started. Minimal output.
Poor (0-2)	0-2	Cannot load data properly. No meaningful algorithm implementation.
3. Code Quality (4 points)		
Excellent (4)	4	Clean code organization. Meaningful variable/function names. Proper indentation. Comments on complex parts. No critical bugs. Follows Python conventions.
Good (3)	3	Generally clean code. Some naming could be better. Basic comments. Few bugs.
Satisfactory (2)	2	Code works but messy. Poor naming. Few comments. Several bugs.
Poor (0-1)	0-1	Very messy code. No comments. Many bugs. Hard to understand.

Part B: Progress Documentation (10 points)

Criterion	Points	Description
1. Progress Summary (3 points)		
Excellent (3)	3	Clear summary of what is complete. Honest assessment of percentage done. Detailed status of each component.
Good (2)	2	Summary provided but lacking detail. Percentage estimate reasonable.
Satisfactory (1)	1	Vague summary. Unclear percentage. Limited component status.
Poor (0)	0	No clear summary provided.
2. Technical Details (4 points)		

Criterion	Points	Description
Excellent (4)	4	Detailed explanation of implemented data structures. Algorithm description clear. Time complexity analysis correct. Design decisions justified.
Good (3)	3	Good explanation of structures. Algorithm description adequate. Some complexity analysis. Reasonable design rationale.
Satisfactory (2)	2	Basic description of structures. Limited algorithm detail. Minimal complexity analysis.
Poor (0-1)	0-1	Vague technical details. No complexity analysis. Poor explanation.
3. Challenges & Solutions (2 points)		
Excellent (2)	2	Specific problems described. Solutions attempted. Current blockers identified with plans to resolve.
Good (1.5)	1.5	Some challenges mentioned. Basic solutions described.
Satisfactory (1)	1	Challenges mentioned but vague. Limited solutions.
Poor (0)	0	No challenges discussed or unrealistic assessment.
4. Remaining Work (1 point)		
Excellent (1)	1	Clear list of remaining tasks. Updated realistic timeline. Confident of completion.
Good (0.75)	0.75	Remaining work identified. Timeline updated.
Satisfactory (0.5)	0.5	Vague remaining work. Unrealistic timeline.
Poor (0)	0	No plan for remaining work.

MILESTONE 3: Final Submission (60 Points)

Due: January 19, 2026 (Week 15)

Part A: Code Implementation (30 points)

Criterion	Points	Description
1. Correctness & Completeness (12 points)		
Excellent (11-12)	11-12	All features work correctly. Algorithm produces accurate results. Handles all test cases including edge cases. No crashes. Meets all requirements from proposal.
Very Good (9-10)	9-10	Most features work. Accurate on standard cases. Minor issues with edge cases. One or two minor bugs.
Good (7-8)	7-8	Core features work. Generally accurate. Some edge cases fail. A few bugs present.
Satisfactory (5-6)	5-6	Basic functionality works. Accuracy questionable. Many edge cases fail. Several bugs.
Poor (0-4)	0-4	Incomplete features. Inaccurate results. Crashes frequently. Major bugs.
2. Data Structure Implementation (10 points)		
Excellent (9-10)	9-10	All data structures built from scratch. No forbidden libraries. Structures are efficient and appropriate. Operations implemented correctly. Integration seamless.
Very Good (7-8)	7-8	Data structures custom-built. Mostly efficient. Good operation implementation. Integration works well.
Good (5-6)	5-6	Data structures implemented but some use built-in features inappropriately. Reasonable efficiency. Integration acceptable.
Satisfactory (3-4)	3-4	Some data structures custom, others use forbidden libraries. Limited efficiency. Poor integration.
Poor (0-2)	0-2	Relied heavily on forbidden libraries. Minimal custom implementation. Inefficient or incorrect.
3. Code Quality (5 points)		
Excellent (5)	5	Exceptionally clean code. Excellent organization (classes, modules). Perfect naming conventions. Well-commented. Consistent style. Professional quality.
Very Good (4)	4	Very clean code. Good organization. Clear naming. Adequate comments. Consistent style.
Good (3)	3	Clean code. Reasonable organization. Acceptable naming. Some comments. Mostly consistent.
Satisfactory (2)	2	Functional but messy. Poor organization. Confusing names. Few comments. Inconsistent style.
Poor (0-1)	0-1	Very messy. No organization. Bad naming. No comments. No consistent style.
4. Testing (3 points)		
Excellent (3)	3	Comprehensive test suite. Tests all major functions. Includes edge cases. Good test coverage (70%+). Test results documented.
Good (2)	2	Good test coverage. Tests main functions. Some edge cases. Results shown.
Satisfactory (1)	1	Basic tests present. Limited coverage. Few edge cases.
Poor (0)	0	No systematic testing. Only manual tests. No edge cases.

Part B: Documentation (15 points)

Criterion	Points	Description
1. Code Comments (3 points)		
Excellent (3)	3	Every class and function has docstring. Complex logic explained. Inline comments where needed. Comments are helpful and accurate.
Good (2)	2	Most functions documented. Some complex parts explained. Comments generally helpful.
Satisfactory (1)	1	Basic comments present. Many functions undocumented. Limited explanation.
Poor (0)	0	No or very few comments. No docstrings.
2. README File (3 points)		
Excellent (3)	3	Perfect README. Clear setup instructions. All dependencies listed with versions. Multiple usage examples. Troubleshooting section. Expected output shown.
Good (2)	2	Good README. Setup instructions clear. Dependencies listed. Basic usage example.
Satisfactory (1)	1	Basic README. Minimal instructions. Some dependencies listed. Unclear usage.
Poor (0)	0	No README or very incomplete. Cannot run code from README.
3. Technical Report (9 points)		
3a. Problem & Motivation (1 point)		
Excellent (1)	1	Clear problem statement. Well-motivated. Background provided.
Good (0.75)	0.75	Problem clear. Basic motivation.
Satisfactory (0.5)	0.5	Problem stated but unclear motivation.
Poor (0)	0	Vague problem. No motivation.
3b. Architecture & Design (2 points)		
Excellent (2)	2	Excellent system architecture diagram. All components explained. Design decisions justified. Clear data flow.
Good (1.5)	1.5	Good architecture explanation. Basic diagram. Most design choices justified.
Satisfactory (1)	1	Architecture described but unclear. Poor or missing diagram. Limited justification.
Poor (0)	0	No architecture description. No diagram.
3c. Data Structures & Algorithms (2 points)		
Excellent (2)	2	Thorough explanation of all data structures. Algorithm description clear. Implementation details provided. Pseudocode included.
Good (1.5)	1.5	Good explanation. Algorithm clear. Some implementation details.
Satisfactory (1)	1	Basic explanation. Algorithm vaguely described. Few details.
Poor (0)	0	No clear explanation. Algorithm unclear.
3d. Complexity Analysis (2 points)		
Excellent (2)	2	Correct time complexity for all major operations. Space complexity analyzed. Big-O notation used correctly. Analysis justified with reasoning.
Good (1.5)	1.5	Mostly correct complexity. Some analysis provided. Big-O used.

Criterion	Points	Description
Satisfactory (1)	1	Complexity mentioned but incorrect or unjustified. Partial analysis.
Poor (0)	0	No complexity analysis or completely wrong.
3e. Results & Analysis (2 points)		
Excellent (2)	2	Comprehensive experimental results. Performance benchmarks on multiple datasets. Comparison with baseline. Results visualized well. Analysis insightful.
Good (1.5)	1.5	Good results presented. Benchmarks provided. Some comparison. Basic visualization.
Satisfactory (1)	1	Basic results. Limited benchmarking. Poor visualization. Shallow analysis.
Poor (0)	0	No results or unconvincing. No benchmarking. No visualization.

Part C: Performance & Analysis (10 points)

Criterion	Points	Description
1. Efficiency (4 points)		
Excellent (4)	4	Runs very fast on large datasets (1000+ samples). Memory efficient. Handles scale well. No performance issues.
Good (3)	3	Runs in reasonable time. Handles moderate datasets well. Acceptable memory usage. Minor performance issues.
Satisfactory (2)	2	Runs slowly but completes. Struggles with large datasets. Memory usage high. Noticeable performance problems.
Poor (0-1)	0-1	Very slow or crashes on large datasets. Memory issues. Cannot handle scale.
2. Complexity Analysis (3 points)		
Excellent (3)	3	All time complexities correct and justified. Space complexity analyzed. Big-O notation perfect. Mathematical reasoning provided.
Good (2)	2	Most complexities correct. Reasonable justification. Big-O mostly correct.
Satisfactory (1)	1	Some complexity analysis. Errors present. Limited justification.
Poor (0)	0	No or completely wrong complexity analysis.
3. Benchmarking (3 points)		
Excellent (3)	3	Comprehensive benchmarks. Multiple metrics (time, memory, accuracy). Comparison with sklearn/baseline. Results well-visualized (charts). Statistical significance tested.
Good (2)	2	Good benchmarks. Main metrics provided. Some comparison. Basic visualization.
Satisfactory (1)	1	Basic benchmarks. Limited metrics. Minimal comparison. Poor visualization.
Poor (0)	0	No benchmarking or unconvincing results.

Part D: Presentation (5 points)

Criterion	Points	Description
1. Demo Quality (2 points)		

Criterion	Points	Description
Excellent (2)	2	Smooth, professional demo. System works perfectly. Shows all key features. Well-paced (10 min). Good slides.
Good (1.5)	1.5	Good demo. System works. Shows main features. Timing good. Decent slides.
Satisfactory (1)	1	Demo works but rough. Some technical issues. Rushed or too slow. Basic slides.
Poor (0)	0	Demo fails. System doesn't work. Poor preparation. Bad slides.
2. Technical Explanation (2 points)		
Excellent (2)	2	Crystal clear explanation of approach. Design decisions well-justified. Complexity discussed confidently. Shows deep understanding.
Good (1.5)	1.5	Clear explanation. Most decisions justified. Basic complexity discussion. Good understanding shown.
Satisfactory (1)	1	Explanation adequate but unclear at times. Limited justification. Shallow understanding.
Poor (0)	0	Unclear explanation. No justification. Little understanding shown.
3. Q&A (1 point)		
Excellent (1)	1	Answers all questions confidently and correctly. Shows expert understanding. Addresses concerns well.
Good (0.75)	0.75	Answers most questions correctly. Shows good understanding.
Satisfactory (0.5)	0.5	Struggles with some questions. Limited understanding.
Poor (0)	0	Cannot answer questions. No understanding shown.

Penalty Deductions

Points may be deducted for the following violations:

Violation	Penalty
Late submission (per day)	-50% of milestone points
Code does not run	-20 points
Missing README	-5 points
Using forbidden libraries	-15 points
Incomplete submission	-10 points
Poor code organization	-5 points
No comments	-5 points
Plagiarism detected	-100 points (entire project = 0)

Bonus Points (Maximum +5)

Exceptional work may earn bonus points:

Achievement	Bonus
Outstanding code quality	+2 points
Exceptional performance optimization	+2 points
Excellent visualization/UI	+2 points
Creative solution to hard problem	+3 points
Published code (GitHub with documentation)	+2 points
Demo video included	+1 point

Note: Total score cannot exceed 100 points

Final Grade Calculation

Project score = Proposal + Progress + Final - Penalties + Bonuses

Maximum: 100 points

This score is then scaled to 25% of your final course grade:

Course Grade Impact = (Project Score / 100) × 25%

Example:

- If you score 85/100 on project: 85/100 × 25% = 21.25% of course grade
- If you score 92/100 on project: 92/100 × 25% = 23% of course grade

---

Questions about grading?
Contact your instructor or TA.

Good luck with your project!