# Examination and Evaluation Methods and Reforms

The evaluation process at IIIT Hyderabad is aligned with outcome-based education and is more in line with testing of concepts and applications rather than recall based questions. This document is to be used as a guideline to understand evaluation methods for various activities and courses.

**Written Examinations:** The written examination-based process involves two short duration quiz exams, one medium duration mid-semester exam, and a long duration final examination. The quiz exams usually test the ability of the student to remember and understand key concepts from lectures and homework sets. The mid-semester and final examinations contain questions that go to level three of the Bloom's taxonomy and require students to apply concepts and test the ability of the students to apply and analyze their learning to new and unstructured problem solving. Question papers can be prepared keeping these objectives in mind in addition to ensuring that the question papers have questions that test students on all the course outcomes.

To ease the process of grading, especially in large classes, the following are **some best-practices** that can be made use of as appropriate.

- Use the centralized grading room so as to minimize moving the answer booklets and thereby reduce the risk of misplacement. This facility is created around the examination schedules and separate emails will be sent for information.
- Prepare a rubric for grading each question of the exam. Prepare guidelines on partial credit. These guidelines make it easy to keep the grading process fair and transparent.
- Keep the scores on each question in addition to the overall score on the exam. This helps in analyzing how students fared on a particular question or topic taught in the class.

**Course Projects:** The presence of a project as part of the evaluation in most courses including core undergraduate courses allows students to achieve outcomes that rely on self-learning, ability to work in a team, ability to communicate, design solutions for complex engineering problems, use of tools, project management, and the like. Working on course projects usually makes the students achieve higher levels of taxonomy on the Blooms taxonomy chart such as Analyzing, Evaluating, and Creating. In view of the advanced nature of the course work at IIITH, it is possible that course projects in advanced electives can constitute a higher component of the overall grade. In core courses, and early undergraduate courses, it is recommended that the contribution of the course project to the overall grade is kept on the lower side. This is subject to the observation that most students in early undergraduate courses do not have a fair and complete idea of how to execute a project.

Projects are to be evaluated by the course faculty on an individual basis for each team on metrics such as the objectives and outcomes, presentation and reports, results obtained, and technical depth. The following rubric may be adopted for the purpose of evaluation. The document in Appendix A, or some modification thereof, can be used based on the level of the course and the expectations of the course project.

| Parameter | Excellent (5) | Very Good (4) | Good (3) | Moderate (2) | Needs Improvement (1) |
|---|---|---|---|---|---|
| Working/ Functional Code | | | | | |
| Observations/ Plots/ Datasets | | | | | |
| Report | | | | | |
| Presentation | | | | | |
| Overall Results | | | | | |
| Comments | Textual comments justifying the scores above | | | | |

**Project/Independent-Study based Credits:** Students that fit into the category of advanced learners can earn some credits via projects or independent study credits. For these credits, students usually sign up with an advisor and together list the outcomes to be achieved at the end of the project or independent study. During the semester, the student and the advisor meet at least once a week to set specific goals to pursue in the coming week. At the end of the semester, the student prepares a report outlining the learning of the activity. The faculty advisor reads the report, agrees on what outcomes are achieved, and assign a letter grade to the student. For more operational details, the reader is referred to the policy at http://intranet.iiit.ac.in/offices/static/files/ProjectIndependent_Study_Guidelines.pdf.

The above rubric can be adopted to justify the grade assigned.

**Appendix A**
## General Instructions for Project Submission[1]


- On Moodle, submit your presentation. Without the submission, you will not be allowed to present and will get an automatic 0. **Deadline: <DATE> <TIME>.** Submit well before time. Emails with reasons such as the Submission Server not accessible will not be entertained.
- The presentation will be downloaded and run from a common laptop/machine. Only demos associated with the presentation will be allowed to run from your personal laptops.
- The presentation should include a link to the GitHub repo (see 'General Instructions for Presentation' below).
- The Github repository should contain clear documentation. In the documentation, include all the necessary details required for us to download the git repo and run your code. If your code does not run on our laptop/machine, you will lose all marks. Test your code by downloading a copy of the repository as a zip file [https://stackoverflow.com/a/2751270] onto a machine different from your laptop/desktop and make sure that the code runs and the documentation describes the procedure to run the code correctly. Plagiarized code will result in an automatic 0.
- You will also need to make sure any data you use/require is included. If it is a publicly available dataset, include documentation instructions for how to download the dataset. If the dataset has been placed in a cloud storage location, provide a working link (GDrive, Dropbox). Mention where to place the data within the directory structure for your project.
- We will be evaluating your code/Github repository before the presentations. Any changes after the deadline (**<DATE> <TIME>**) will not be considered and will be penalized. Remember that TAs are members of your code repo, so no timestamp rollbacks, please.

## General Instructions for Presentation

- Title Slide: Include Team Name, Team Members (Roll No. and Program), Title of the Project, Mentor TA, Repo URL
- End Slide: Clear details about division of work (who did what).
- Regarding the division of work: Stating that "everyone worked on everything equally" is not an acceptable (or realistic) division. We will be looking at your GitHub repo's Insights tab and the related analytics [https://github.blog/2012-04-25-introducing-the-new-github-graphs/]. Any abnormal trend (e.g. Commits only by a single person, the majority of commits happening very close to project deadline) will be penalized.
- Begin by introducing yourself and the team. The presentation will stop at the 12th minute for each team. The remaining 3 minutes will be for VIVA and demo. Pick a presentation format: Either one person presents from the team or all present. Rehearse for 12 minutes. Time will be maintained strictly. Failure to finish within the time will result in a deduction of marks.
- Plan to give a live DEMO of your project. If it will take time, run the program before the presentation starts so that it is ready by the end of the presentation.
- Every team should be present at least 5 minutes before the start of their slot.

---

[1] Based on a document designed by Dr Ravi Kiran, IIIT Hyderabad.

**Grading Rubric (on a scale of 350)**:

- Repo [50]
  - Documentation, Code, Presentation slides included [30]
  - No abnormal commit trends [20]
- Code [60]
  - Runs as per documentation [50]
  - Commented [10]
- Documentation  [40]
  - Detailed and clear enough to run stand-alone without any errors
- Presentation
  - Slides [50]
    - Opening and Ending Slide as per instructions [10]
    - Neat, avoids death-by-powerpoint [15] (Watch: https://www.youtube.com/watch?v=KbSPPFYxx3o)
    - Includes results other than standard datasets. Show results on data specifically collected by project member(s) [25]
  - Delivery [100]
    - All team members present [20]
    - Completed in time [30]
    - Demo [50]
  - Viva [50]