

Project Title: - Python-Based CGPA & GPA Calculator

📖 Overview of the Project

This project is a simple command-line Python application that helps students calculate their Semester GPA (and overall CGPA if extended).

The program prompts the user to enter the number of courses and provides two options for entering marks:

1. Enter total marks directly, or
2. Enter marks separately for Mid Sem, Internals, and End Sem.

Using this input, the calculator validates the marks, converts them into grade points, multiplies them by course credits, and finally computes the GPA for the semester.

✨ Features

- ✓ User-friendly command-line interface
- ✓ Supports two mark entry modes (Total marks OR individual components)
- ✓ Performs automatic grade point calculation using a ceiling-based formula
- ✓ Calculates GPA accurately using weighted credits
- ✓ Input validation to prevent invalid marks
- ✓ Modular code structure with a dedicated GPA calculation function

🛠 Technologies / Tools Used

- Programming Language: Python 3
- Libraries Used:
- math (for ceiling function)
- Environment: Any Python-supported terminal (Windows / macOS / Linux)

📥 Steps to Install & Run the Project

1 Install Python (if not installed)

Download Python from the official website:

<https://www.python.org/downloads/>

2 Clone or Download the Project

git clone <https://github.com/your-repo/cgpa-calculator.git> or simply copy the script into a .py file (e.g., cgpa_calculator.py).

3 Run the Program

Open a terminal and execute: `python cgpa_calculator.py`

4 Follow On-Screen Instructions

- Enter the number of courses
- Choose how you want to enter marks
- Enter course details & marks
- View your calculated GPA

Instructions for Testing

To properly test the project:

Test Case 1: Total Marks Input

1. Run the program
2. Enter 3 for number of courses
3. Select 0 (enter total marks)
4. Enter course name, credits, and marks

5. Verify the GPA output

Test Case 2: Component-Based Marks Input

1. Run the program
2. Enter any number of courses
3. Select 1 (enter Mid Sem + Internals + End Sem)
4. Enter all components
5. Check if the sum & GPA are correct

Test Case 3: Invalid Marks

- Enter marks greater than 190 to ensure the system shows “Enter Valid Marks”.

🛠️ Problem Statement

Manually determining GPA or CGPA can be an error-susceptible task, for students. Since various subjects carry credit values and employ diverse grading methods (Mid Sem, Internals, End Sem) precisely translating marks into grade points and calculating the final GPA becomes challenging.

This initiative addresses the issue by offering an automated GPA calculator written in Python which takes scores in two formats and calculates the GPA methodically guaranteeing precision and uniformity.

🎯 Scope of the Project

The project concentrates on calculating the Semester GPA while allowing for the expansion of the method to include CGPA computation.

The extent encompasses:

* Enabling users to input scores either as marks or broken down by components (Mid Sem, Internals, End Sem).

- Validating the entered marks.
- Converting marks into grade points using a ceiling-based formula.
- Multiplying grade points by credit weightage to calculate final GPA.
- Displaying the computed GPA clearly to the user.

This project is designed as a terminal application and has the potential to be extended for academic utilities, departmental platforms or student information systems.

👤 Target Users

This project is ideal for:

- Students seeking an dependable method to calculate their GPA.
- Educators / Faculty members requiring a calculator for assessment.
- Educational institutions looking for a basic script for internal tools.
- Novice Python users aiming to grasp how to manage user inputs utilize functions, implement loops and perform calculations.

✨ High-Level Features

- ✓ Dual Methods, for Entering Marks

Users have the option to input either the marks or the individual components (Mid Sem + Internal + End Sem).

- ✓ Automatic Grade Calculation

Uses `math.ceil(marks / 19)` to convert marks to grade points.

- ✓ Accurate GPA Calculation

Computes GPA based on weighted credits for each course.

- ✓ Input Validation

Discards scores (, over 190) and guarantees proper input processing.

- ✓ Modular Function Architecture

The primary GPA calculation logic resides in a function called `calculate_gpa()` keeping the code structured and straightforward to update.

- ✓ Works on Any Terminal

Fully compatible with Windows, macOS, and Linux Python environments.