# 03. Student Credits

*Diyan is a student and he wants your help.*

*He wants a program that calculates whether he gets a diploma or not.*

Write a function **students\_credits** which receives a different number of strings.

Each string will be in the format: **"{course name}-{credits}-{max test points}-{diyan's points}"**.

Your task is to **calculate the credits Diyan manages to get from all courses**. The credits he gets are **proportional to his points on the test**. For example, if the **credits of a course are 25**, and Diyan **achieved to get 50 of 100 max test** **points**, he will get **12.5 credits** for the course.

Also, you need to **keep track of each course** and **Diyan's credits** and **sort them in** **descending order** by **Diyan's credits.**

Finally**, return a string on multiple lines** containing**:**

* Diyan's achievement message:
  + If the sum of all of Diyan's credits is **more than or equal to 240**, return: **"Diyan gets a diploma with {total credits} credits."**
  + Otherwise, return: **"Diyan needs {credits needed} credits more for a diploma."**
* Information for **each course** and **Diyan's credits:**
* **"{course name} - {diyan's credits}"**
* **Note: Each course data should be on a new line.**
* All **credits** must be **formatted to the first decimal place.**

***Note: Submit only the function in the judge system***

### Input

* There will be **no input**, just any number of strings with courses data passed to your function

### Output

* The function should **return a string** in the format described above:

### Constraints:

* There will always be **at least** **one course**.
* There will **not** be two or more courses with the **same name**.
* All points and all credits will be **whole numbers**

### Examples

|  |  |
| --- | --- |
| **Test Code** | **Output** |
| print(  students\_credits(  "Computer Science-12-300-250",  "Basic Algebra-15-400-200",  "Algorithms-25-500-490"  )  ) | Diyan needs 198.0 credits more for a diploma.  Algorithms - 24.5  Computer Science - 10.0  Basic Algebra - 7.5 |
| **Comment** | |
| First, we get the data for the **Computer Science** course. The total credits for this course are 12, and Diyan manages to reach 250 points out of 300 total points on the test. We calculate what percentage of the test Diyan took -> **250 / 300 = 83.3%**. After that, we find the credits that he has for this course -> **12 \* 83.3% = 10**.  Next, we get the data for the **Basics Algebra** course. Diyan manages to get **200/400 points** on the test and receives **7.5 credits**.  We get the data for the **Algorithms** course. Diyan manages to get **490/500 points** on the test and receives **24.5 credits**.  Diyan's **total credits are 42**. However, it is **less than 240**, so he can't get a diploma.  Finally, we **sort the courses** by Diyan's credits in **descending order** and **return** all the **needed output**. | |
| print(  students\_credits(  "Discrete Maths-40-500-450",  "AI Development-20-400-400",  "Algorithms Advanced-50-700-630",  "Python Development-15-200-200",  "JavaScript Development-12-500-480",  "C++ Development-30-500-405",  "Game Engine Development-70-100-70",  "Mobile Development-25-250-225",  "QA-20-300-300",  )  ) | Diyan gets a diploma with 243.3 credits.  Game Engine Development - 49.0  Algorithms Advanced - 45.0  Discrete Maths - 36.0  C++ Development - 24.3  Mobile Development - 22.5  AI Development - 20.0  QA - 20.0  Python Development - 15.0  JavaScript Development - 11.5 |
| print(  students\_credits(  "Python Development-15-200-200",  "JavaScript Development-12-500-480",  "C++ Development-30-500-405",  "Java Development-10-300-150"  )  ) | Diyan needs 184.2 credits more for a diploma.  C++ Development - 24.3  Python Development - 15.0  JavaScript Development - 11.5  Java Development - 5.0 |