NATIONAL UNIVERSITY
OF COMPUTER & EMERGING SCIENCES
PESHAWAR CAMPUS

Problem Set:	Assignment: AG04	**Semester**:	Fall 2017
Points:	*See autograder*		
Date Set:	*See autograder*	**Due Date**:	*See autograder*
Course:	CS101 Introduction to Computing	**Instructor**:	Dr. Nauman

1 NUCES Grades and SGPA

Since you are reading this, you have already downloaded and extracted the zip file.

1.1 Tasks to do

1. Open the file `a04.py` and look between the markers. You may ignore the code outside the markers completely.
   You may run the code by typing the following from the shell: `python a04.py`
   This will not run the tests but the code itself.
2. There are two main tasks to complete.
   (a) Write a function with the name `get_grade` the takes in the total marks as input and calculates the grade according to FAST NUCES standard. This standard is given in the following table:

Grade	Given when marks are at least	Point Equivalent
A+	90	4.00
A	86	4.00
A-	82	3.67
B+	78	3.33
B	74	3.00
B-	70	2.67
C+	66	2.33
C	62	2.00
C-	58	1.67
D+	54	1.33
D	50	1.00
F	0	0.00

   Note that according to the rules, the total points scored by the student are *rounded* to the nearest integer before assigning them a grade.

   (b) The second function you need to write is `calculate_sgpa`. This function takes in three inputs – each input representing a grade received in a subject – and calculate the Semester GPA (SGPA) based on the point equivalent of the grades. For instance, if the function is called with the grades 'A', 'A' and 'D', it will return SGPA of **3.0** since $(4+4+1)/3 = 3.0$.

   However, this function also allows the caller to calculate SGPA of less than three subjects. The way to do this is to send the value `'nothing'` as one or two of the arguments. For example, if the function is called with the values 'A', 'B', 'nothing', it should return **3.5**.

Algorithm 1 SGPA Calculation

```
1:  procedure CALCULATE_SGPA(grade1, grade2, grade3)
2:      total_marks ← 0
3:      total_number_of_subjects ← 0
4:      if grade1 does not equal 'nothing' then
5:          increment total_number_of_subjects
6:          increase total_marks according to the point equivalent of grade1 (see table for point equiv.)
7:      if grade2 does not equal 'nothing' then
8:          increment total_number_of_subjects
9:          increase total_marks according to the point equivalent of grade2 (see table for point equiv.)
10:     if grade3 does not equal 'nothing' then
11:         increment total_number_of_subjects
12:         increase total_marks according to the point equivalent of grade3 (see table for point equiv.)
13:     if total_number_of_subjects is 0 then
14:         return 0 as answer
15:     SGPA ← total_marks / total_number_of_subjects
        return SGPA
```

The exact method for calculation of SGPA is given in the Algorithm 1. In this algorithm, the symbol ← is the assignment operator and 'increment' means to increase the value of a variable by one.

3. You may change the values in function calls at the end of the file `a04.py` to check the functions.
4. Run local tests and if they pass, submit the assignment using the submission command given on the Autograder assignment page. (Same as the first assignment.)