Graded Programming Assignment 2 | Graded Programming Assignment (GPA 02) | CS101.1x Courseware

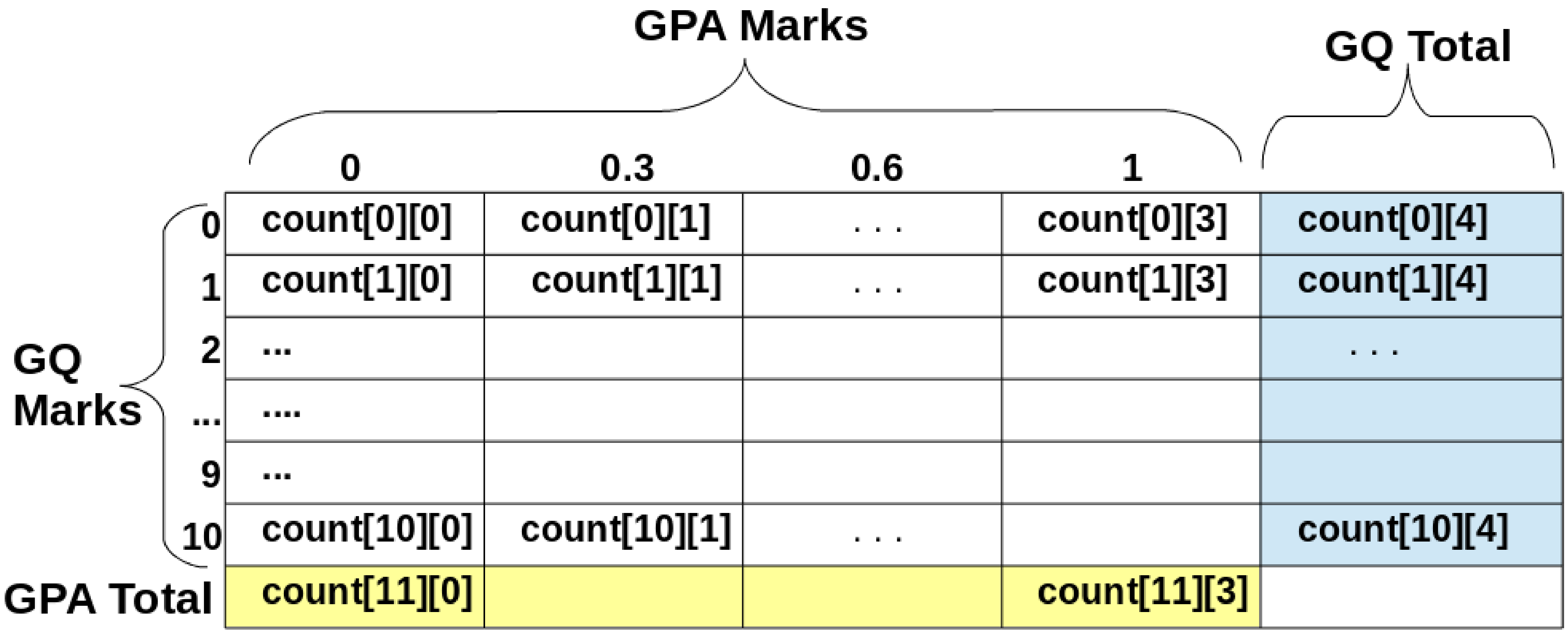
PreviousNext**Graded Programming Assignment (GPA 02 - 10 Marks)Overview of the Problem**
CS101.1x: The CS101.1x course conducts some graded quizzes and graded programming assignments for the students to solve. Let us consider one such case, where 1 Graded Quiz and 1 Graded Programming Assignment is conducted so far.

GQ (Graded Quiz): The quiz contains 10 questions of 1 mark each i.e. entire quiz is of 10 marks, where students can score from 0 (lowest) to 10 (highest)

GPA (Graded Programming Assignment): It is of 10 marks. Students need to write some code and submit it. There are 3 test cases, against which the student's program is checked, and the marks are awarded. If the student's submission passed all three test cases, then 1 mark is awarded; if two testcases are passed then, 0.6 marks are awarded; if only one test case is passed, then 0.3 marks are awarded; and finally 0 marks are awarded, if no testcase passes. This means that the student can score from 0 (lowest), 0.3, 0.6, and 1 (highest).

Problem Definition
Assume that the students have successfully answered both, the GQ and GPA. We now need to analyze the scores. i.e. we want to know the following:-
(a) how many students scored 10 marks in GQ, how many scored 9 marks in GQ and so on, i.e. 'x' marks in GQ (See last column of the table).
(b) how many students scored 1 mark in GPA, how many scored 0.6 marks in GPA and so on, i.e. 'y' marks in GPA (See last row of the table).
(c) A combination of both of these,
i.e. we want to know how many students scored 10 marks in GQ and the same set of students scoring 0 marks in GPA; how many scored 4 marks in GQ and 0.6 marks in GPA, etc.
In short we want to know how many students scored 'x' marks in GQ and same students scoring 'y' marks in GPA. This can best be represented using a matrix. See the example given below, which shows the array 'count[][]' and how it is accessed.

The array contains number of students scoring 'x' in GQ, number of students scoring 'y' in GPA, and a combination of both of these.



		GPA Marks				GQ Total
		0	0.3	0.6	1	
GQ Marks	0	count[0][0]	count[0][1]	. . .	count[0][3]	count[0][4]
	1	count[1][0]	count[1][1]	. . .	count[1][3]	count[1][4]
	2	...				. . .
	...	....				
	9	...				
	10	count[10][0]	count[10][1]	. . .		count[10][4]
GPA Total		count[11][0]			count[11][3]	

Overview of the variables used in the main() Program already written (which is given here for your reference)

1. Total number of students, say 'n', stored in variable '**numOfStudents**'
2. Roll Number of 'n' students, stored in array '**roll[]**'
3. Marks scored by 'n' students in GQ, stored in array '**gq[]**'
4. Marks scored by 'n' students in GPA, stored in array '**gpa[]**'
5. This means that roll[index], gq[index], and gpa[index] contains roll number, marks of GQ, and marks of GPA of a student. Here 'index' denotes a student.

You are required to write 3 functionsFunction 1: getTotalGQ *This function has 3 parameters:*

1. **int numOfStudents**: Integer number containing the total number of students
2. **int gq[]**: Integer array containing the GQ marks of all students
3. **int sum_gq[]:** Integer array to be computed

The function should do the following

1. Count the number of students who scored 'x' marks in GQ.
2. Store the result in the array, 'sum_gq[]'.

   E.g. 1: If 45 students scored 9 marks in GQ, then sum_gq[9] = 45

   E.g. 2: If 34 students scored 0 marks in GQ, then sum_gq[0] = 34

You need to write your code in this function

```
void getTotalGQ(int numOfStudents, int gq[], int sum_gq[]) {


}
```

Function 2: getTotalGPA *This function has 3 parameters:*

1. **int numOfStudents:** Integer number containing the total number of students
2. **float gpa[]:** Array of type float containing the GPA marks of all students
3. **int sum_gpa[]:** Integer array to be computed

The function should do the following

1. Count the number of students who scored 'y' marks in GPA.
2. Store the result in the array, 'sum_gpa[]'.

   E.g. 1: If 20 students scored 1 mark in GPA, then sum_gpa[3] = 20

   E.g. 2: If 10 students scored 0.6 marks in GPA, then sum_gpa[2] = 10

   E.g. 3: If 19 students scored 0.3 marks in GPA, then sum_gpa[1] = 19

   E.g. 4: If 42 students scored 0 marks in GPA, then sum_gpa[0] = 42

You need to write your code in this function

```
void getTotalGPA(int numOfStudents, float gpa[], int sum_gpa[]) {


}
```

Function 3: getTotalCount *This function has 3 parameters:*

1. **int numOfStudents**: Integer number containing the total number of students
2. **int gq[]:** Integer array containing the GQ marks of all students
3. **float gpa[]:** Array of type float, containing the GPA marks of all students
4. **int count[][]**: Integer array to be computed

The function should do the following

1. Count the number of students who scored 'x' marks in GQ and 'y' marks in GPA. Store the result in the array 'count[][]'.
2. Count the number of students who scored 'x' marks in GQ and store in count[x][4], where 'x' ranges from 0 to 3. Hint: Add count[0][0] to count[0][3].
3. Count the number of students who scored 'y' marks in GPA and store in count[11][y], where 'y' ranges from 0 to 10. Hint: Add count[0][0] to count[10][0]
4. **E.g. 1:** if gq[index] = 8 and gpa[index] = 0.3, then increment the value of array count[8][1], by 1. This means that value at count[8][1] are number of students who scored 8 marks in GQ and same set of students who scored 0.3 marks in GPA.

   E.g. 2: If gq[index] = 2 and gpa[index] = 0, then increment the value of array count[2][0], by 1. This means that value at count[2][0] are number of students who scored 2 marks in GQ and same set of students who scored 0 marks in GPA.

   E.g. 3: If gq[index] = 0 and gpa[index] = 0, then increment the value of array count[0][0], by 1. This means that value at count[0][0] are number of students who scored 0 marks in GQ and same set of students who scored 0 marks in GPA.

You need to write your code in this function

```
void getTotalCount(int numOfStudents, int gq[], float gpa[], int count[][5]) {


}
```

Example

Consider an example given below that contains data for 54 students. Since it is not feasible to display the data for all 54 students, only the initial data is given below. Click here to view the entire data of all 54 students.

roll[i]	gq[i]	gpa[i]

4001	7	0
4002	10	1
4003	10	1
4004	7	0
...	...	...
4054	...	...

Based on the data of all 54 students, the analysis would be as follows:

Marks GQ / GPA	**0**	**0.3**	**0.6**	**1**	**GQ Total**
0	2	3	2	2	9
1	0	0	0	0	0
2	0	0	4	3	7
3	1	1	3	0	5
4	2	1	2	0	5
5	0	1	1	0	2
6	3	0	1	0	4
7	2	0	0	1	3
8	1	2	1	0	4

9	0	3	0	2	5
10	4	1	2	3	10
Total GPA	15	12	16	11	54

This table shows (a) total number of students who scored 'x' in GQ, independent on GPA **(see the last column, which adds the entire row),** (b) total number of students who scored 'y' in GPA, independent on GQ **(see the last row, which adds the entire column),** (c) rest of the matrix that stores the number of students who scored 'x' in GQ and 'y' in GPA.

We can see that there are 4 students who scored 10 marks in GQ and the same students who scored 0 marks in GPA. There are 3 students who scored 9 marks in GQ and the same students who scored 0.3 marks in GPA.
We can see (from the last row of the table), that there are 15, 12, 16, and 11 students scored 0, 0.3, 0.6, and 1 marks respectively in GPA, independent of GQ.

We can also notice (from the last column of the table) that there are 9, 0, 7, 5, 5, 2, 4, 3, 4, 5, 10 students who scored 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 marks respectively in GQ, independent of GPA.

Coming back to our program, **the values of array, 'count [][]', 'sum_gq[]', and 'sum_gpa[]" will be as follows**

`count[i][j]`

`index i/j`	**0**	**1**	**2**	**3**	**4**
0	2	3	2	2	9
1	0	0	0	0	0
2	0	0	4	3	7
3	1	1	3	0	5
4	2	1	2	0	5

`sum_gq[i]`

`i`	`Value`
0	9
1	0
2	7
3	5
4	5

`sum_gpa[i]`

`i`	`Value`
0	15
1	12
2	16
3	11

5	0	1	1	0	2		**5**	2		
6	3	0	1	0	4		**6**	4		
7	2	0	0	1	3		**7**	3		
8	1	2	1	0	4		**8**	4		
9	0	3	0	2	5		**9**	5		
10	4	1	2	3	10		**10**	10		
11	15	12	16	11	54					

Sample Example

```
Number of students: 54
Roll Number and Marks
4001    7       0
4002    10      1
4003    10      1
4004    7       0
4005    4       0.3
 //... Assuming that data of all 54 students is entered by the user
```

```
GQ Marks         Number of Students
   0                     9
   1                     0
   2                     7
   3                     5
   4                     5
   5                     2
   6                     4
   7                     3
   8                     4
   9                     5
   10                    10

GPA Marks        Number of Students
   0                     15
   0.3                   12
   0.6                   16
   1                     11

-------------------------------------------------------
Marks                                            GQ
GQ/GPA          0       0.3     0.6     1       Total
-------------------------------------------------------
  0             2        3       2       2       9
  1             0        0       0       0       0
  2             0        0       4       3       7
  3             1        1       3       0       5
  4             2        1       2       0       5
  5             0        1       1       0       2
  6             3        0       1       0       4
  7             2        0       0       1       3
  8             1        2       1       0       4
  9             0        3       0       2       5
  10            4        1       2       3       10
-------------------------------------------------------
GPA Total       15      12      16      11      54
```

```
-----------------------------------------------------------------
```

Click on the next unit (visible at the top of your screen), to follow the instructions and write your code in the space provided.