Spreadsheet Model for Student Evaluation and Grading

Harshal Lowalekar

Abstract

This technical note describes the application of a spreadsheet-based model for student evaluation and grading. The note shows how a spreadsheet-based model can be used in automating the process of evaluating and grading in situations where advanced technologies like optical recognition are not available. The first part of the note describes how a spreadsheet can be used to evaluate course components, which involve multiple-choice type of questions (MCQs). The second part of the note describes how a spreadsheet can help instructors in determining the final grades for a course and its components, given the institutional grading guidelines. The model was successfully used for several courses at a premier management institute in central India. It was observed that the model helped in saving significant amount of time in student evaluation and grading and thus helped in speeding up the process of feedback from the instructor to the students.

Keywords: spreadsheet model; student evaluation; student grading

1. Introduction

WIMCI[1] is a well-known institute for management education in central India. The institute offers several post graduate and doctoral level programs for fresh graduates and working professionals. The 2-year Post Graduate Programme (PGP) is the flagship programme of the institute and currently admits approximately 450 students, which includes national as well as international students. The first year of the PGP involves compulsory coursework while the second year involves a mix of compulsory and elective courses. At present there are 6 sections (90 students per section) in the first year of the program. A student has to pass a certain minimum number of courses before he/she can be promoted to the second year.

A course typically consists of several components like class participation (CP), quizzes, midterms, end terms, group projects, individual assignments, etc. These course components are given certain weightage while computing the final grade for a given course. The institute follows a relative grading system for evaluating the students in most of the course components. A first year course invariably involves 2 to 3 quizzes along with the midterm and the end term examinations. Due to the large number of students in each section the course instructors often prepare multiple-choice question papers (containing 15 to 30 questions), with multiple sets, for quizzes and examinations. In order to discourage random guessing by the students, a penalty for wrong answers is usually incorporated in such question papers. The help of teaching associates is often taken in evaluating and grading these question papers. However, evaluating 200 to 300 question papers even for multiple-choice questions is a tedious job. It requires a considerable amount of time to tally the answer sheet of a student with the master answer sheet for the given set. The computation of final score for a given student requires manually counting the number of right, wrong and no responses. Since the process of evaluating and grading such huge numbers affects the speed and the quality of feedback to the students, it was felt that some form of automation could be useful in streamlining the process of evaluation and grading at the institute.

Optical mark recognition (OMR) machines have been successfully used in the past at several institutions to resolve some of the issues related to evaluation and grading. However, for institutes which cannot afford an OMR type of a setup due to a variety of reasons, there is an inexpensive technology in the form of spreadsheets, which can be used to achieve results similar to an OMR technology without any additional expenditure. This note demonstrates a simple yet powerful spreadsheet based model, which can help the

1 The actual name of the institute has been changed to WIMCI for the purpose of illustration in this technical note.

instructors in automating the process of grading and evaluation to a great extent.

2. Material and Methods

The next section describes a generic model for evaluating answer sheets for multiple-choice question(MCQ) papers with multiple sets. The subsequent section discusses the model for grading, given the institutional grading guidelines. Microsoft Excel 2000 has been used for the purpose of this note but the model can be developed using many other spreadsheet applications like OpenOffice or Google spreadsheets. The readers can refer to [1] and [2] for an in-depth understanding of the excel-based commands used in the next few sections of this note.

*2.1 Spreadsheet Model for Evaluation**

If the question paper of a quiz or an examination involves multiple-choice kinds of questions the spreadsheet model explained in this section can be used for evaluation purposes. We shall describe the model for a generic situation described below:

- There are 20 questions in the question paper, which are of MCQ kind.
- There are 3 sets of the question paper.
- Each MCQ question has 5 options (A, B, C, D and E) and just one correct option.
- For every correct answer, a student earns 1 mark but for every wrong answer he /she loses one-fourth of a mark. No marks are deducted for not attempting a question.

Three approaches to perform the automatic evaluation are presented. The first method gives the score of the student for each question while the second and third methods just give the total score obtained by the student using array formula and VBA function respectively.

2.1.1 Method 1

The spreadsheet model for the evaluation has been shown in Fig 1.The model functions in the following way:

1. The evaluator enters the set numbers (range B5:B7) and the answer key (range C5:C7) in form of a string of the correct responses, for the respective sets.
2. The marks earned per correct answer and the penalty per wrong answer is entered in the worksheet (Cells H5 and I5).
3. The evaluator enters the roll number (column A, row 16 onwards) of the student along with the student's set number (column B, row 16 onwards) and his responses in a string format (column C, row 16 onwards) in the given sheet. Letter "N" in the response sheet denotes the questions, which a given student did not attempt. The model then compares the string of student's responses with the answer key for that set and computes the total marks earned by the student and displays them in one column (column F). The marks earned by a student for an individual question are also displayed separately (columns G to Z, row 16 onwards). Excel also automatically looks up the name of the student corresponding to the roll number from a student database sheet (Fig 3).
4. Several checks are added to the model to ensure correct entries in the answer keys (range C5:C7), the student responses (column C, row 16 onwards) and the total number of students evaluated (cell C13).

The actual formulae entered in the various cells have been shown in Table 1.The formula entered in cell G16 basically picks up the response for first question from the full string of responses for a given student, compares it with the correct response for the first question of the given set and then returns the marks earned by the student for the same in the cell G16.The same formula is then copied to the other columns (H16 to Z16) using relative referencing. Fig 2 depicts the student database which contains the roll number of the students along with their names and group numbers.(Table 1)

2.1.2 Method 2 (Using Array Formula)

In the previous approach the spreadsheet displays the marks obtained in the individual question. However, using an array formula, it is possible to obtain the total marks obtained by a student in a much cleaner and

* Readers can refer to [3]-[4] for preparing pre-designed MCQ quizzes using google spreadsheets. [5] shows the application of MS-Excel spreadsheet for analyzing the assessment data.

Table 1: Formulae Entered in the Spreadsheet (Method 1)

S. No.	Formula	Entered in Cell(s)	Remark
1	=LEN (C5)	Enter in D5. To be copied in D6:D7	Computes the length of the answer key string.
2	=LEN(C16)	Enter in D16. To be copied in column D, row 17 onwards	Computes the length of the answer key string (Checking for error)
3	=VLOOKUP(A16,'Student Database'!$B$4:$C$48,2, FALSE)	Enter in E16. To be copied in column E, row 17 onwards	Looks up the student name corresponding to the given roll number in 1st column.
4	=IF(G$13<=$D$5, IF (MID ($C16,G$13,1)="N", 0, IF(MID($C16,G$13,1)= MID(VLOOKUP($B16, B$5:$C$8,2,TRUE), G$13,1), 1,$I$5)), " ")	Enter in G16. To be copied in the range G16 to AF100.	Compares the response of 1st question of a given set with the correct answer for the same and returns the marks to be awarded for the question.
5	=SUM(G16:AJ16)	Enter in F16. To be copied in column F, row 17 onwards	Computes the total score for the student

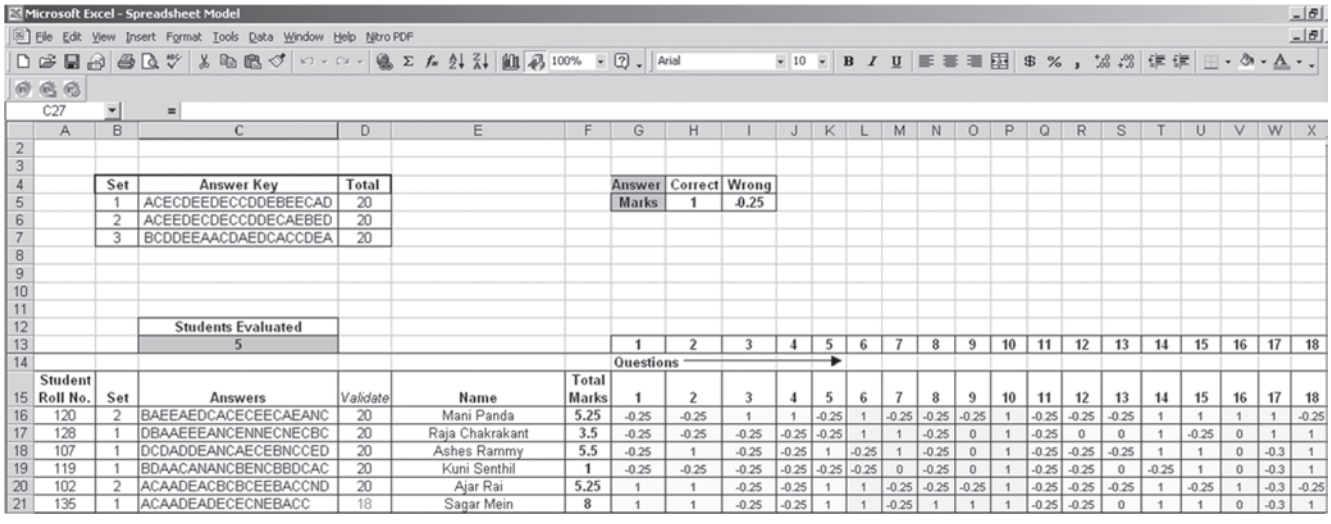


Fig 1: Spreadsheet for Evaluation[2]

2 The students' names have been changed.

	A	B	C	D	E
1					
2					
3		Roll No.	Name	Group	
4		101	Abey Anant	7	
5		102	Ajar Rai	12	
6		103	Ajay Sharma	3	
7		104	Amey Mishra	5	
8		105	Anil Kapoor	15	
9		106	Arvind Das	7	
10		107	Ashes Rammy	2	
11		108	Asin Trivedi	15	
12		109	Barney Ganguly	6	
13		110	Chiral Paun	9	
14		111	Dawy Pillat	13	
15		112	Dean Roy	7	
16		113	Desmond Hansie	4	
17		114	Gurpreet King	15	
18		115	Imrul Kahn	8	
19		116	Jaspreet Ahluwalia	15	
20		117	Jitu Nagesh	2	
21		118	Kamaal Khanum	4	
22		119	Kuni Senthil	1	
23		120	Mani Panda	14	
24		121	Mohsin Patel	13	
25		122	Nilu Phulket	15	
26		123	Prashant Pramod	8	
27		124	Preeti Singania	5	
28		125	Priya Doshi	13	
29		126	Priyanka Sandy	11	
30		127	Rahul Gupt	7	
31		128	Raja Chakrakant	6	
32		129	Rajesh Channy	7	
33		130	Ranbeer Shourya	4	
34		131	Ratul Kureshi	10	

Responses | Group Assignments | Grades | **Student Database**

Fig 2: Student Database[3]

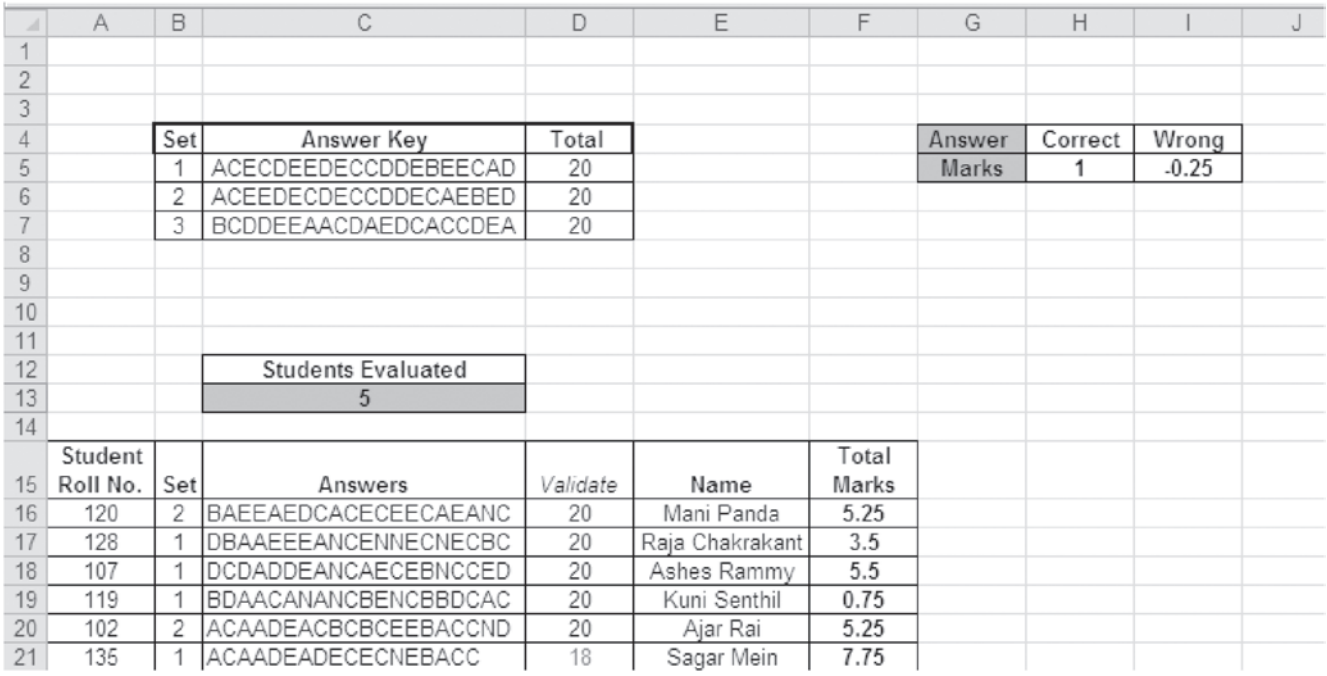
efficient way as shown in Fig 3.The list of commands used for achieving the same are mentioned in Table 2.

2.1.3 Method 3 (Using a VBA function)

Instead of using an array formula to compute the total marks, as discussed in the previous section, one can define a user defined function named "MCQ_evaluator" which will compare the two strings and return the total marks given the marks for a correct and an incorrect answer. The function to be entered in cell F16 (to be copied in column F, row 17 onwards) is shown below:

=MCQ_Evaluator(C16,VLOOKUP(B16,$B$5:$C$7,2,FALSE),$H$5,$I$5)

3. The students' names have been changed.



Set	Answer Key	Total
1	ACECDEEDECCDDEBEECAD	20
2	ACEEDECDECCDDECAEBED	20
3	BCDDEEAACDAEDCACCDEA	20

Answer	Correct	Wrong
Marks	1	-0.25

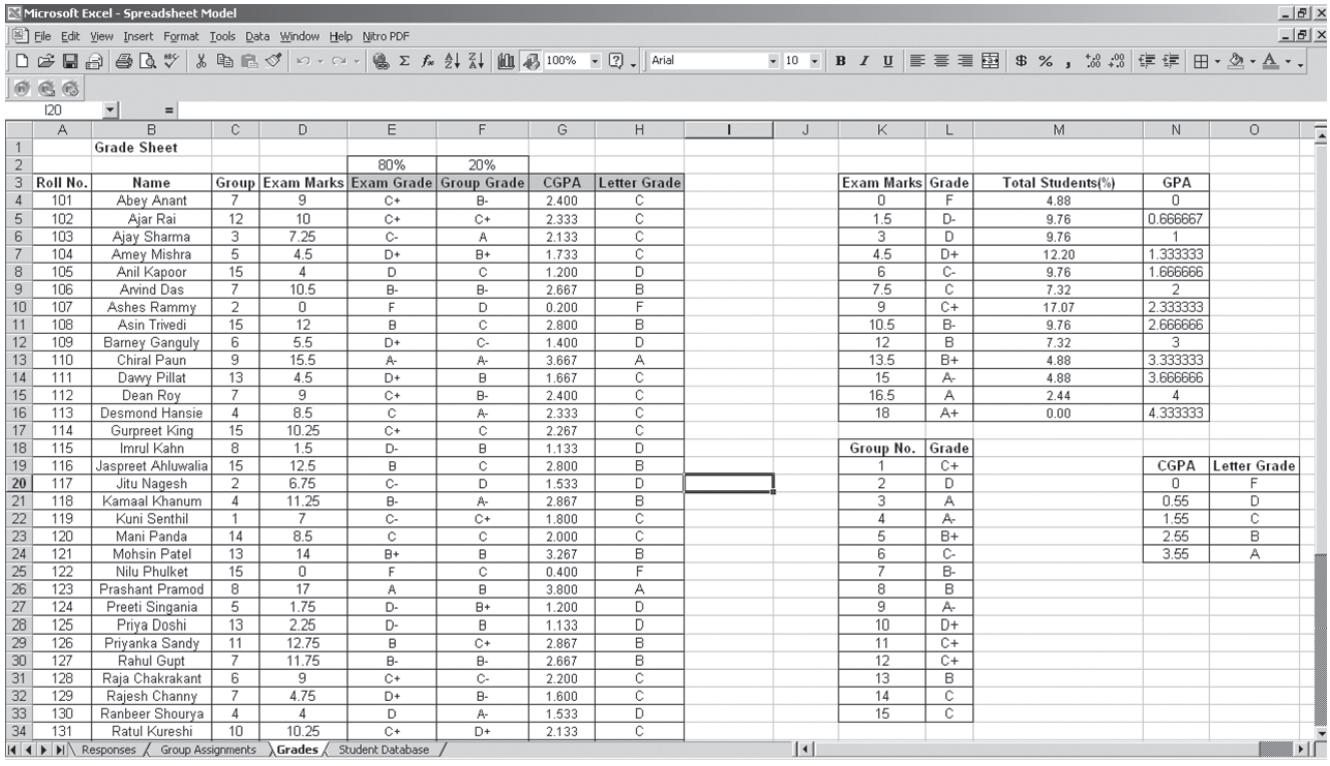
Students Evaluated
5

Student Roll No.	Set	Answers	Validate	Name	Total Marks
120	2	BAEEAEDCACECEECAEANC	20	Mani Panda	5.25
128	1	DBAAEEEANCENNECNECBC	20	Raja Chakrakant	3.5
107	1	DCDADDEANCAECEBNCCED	20	Ashes Rammy	5.5
119	1	BDAACANANCBENCBBDCAC	20	Kuni Senthil	0.75
102	2	ACAADEACBCBCEEBACCND	20	Ajar Rai	5.25
135	1	ACAADEADECECNEBACC	18	Sagar Mein	7.75

Fig 3: Spreadsheet for Evaluation (Methods 2 and 3)

Table 2: Formulae Entered in the Spreadsheet (Using Array Formula)

S. No.	Formula	Entered in Cell(s)	Remark
1	=LEN (C5)	Enter in D5. To be copied in D6:D7	Computes the length of the answer key string.
2	=LEN(C16)	Enter in D16. To be copied in column D, row 17 onwards	Computes the length of the answer key string (Checking for error)
3	=VLOOKUP(A16,'Student Database'!$B$4:$C$48,2, FALSE)	Enter in E16. To be copied in column E, row 17 onwards	Looks up the student name corresponding to the given roll number in 1st column.
4	{=SUM(IF(MID(C16,ROW (INDIRECT("1:"&LEN ($C$5))),1) =MID(VLOOKUP(B16,$B$5: $C$7,2,0), ROW(INDIRECT ("1:"&LEN($C$5))),1),$H$5, IF(MID(C16,ROW(INDIRECT ("1:"&LEN($C$5))),1)= "N",0,$I$5)))}	Ctrl+Shift+Enter in F16. To be copied in column F, row 17 onwards	Compares the responses of all the questions of a given set with the answer key for the same and returns the total marks to be awarded to the student.



Microsoft Excel - Spreadsheet Model

Grade Sheet

Roll No.	Name	Group	Exam Marks	80% Exam Grade	20% Group Grade	CGPA	Letter Grade
101	Abey Anant	7	9	C+	B-	2.400	C
102	Ajar Rai	12	10	C+	C+	2.333	C
103	Ajay Sharma	3	7.25	C-	A	2.133	C
104	Amey Mishra	5	4.5	D+	B+	1.733	C
105	Anil Kapoor	15	4	D	C	1.200	D
106	Arvind Das	7	10.5	B-	B-	2.667	B
107	Ashes Rammy	2	0	F	D	0.200	F
108	Asin Trivedi	15	12	B	C	2.800	B
109	Barney Ganguly	6	5.5	D+	C-	1.400	D
110	Chiral Paun	9	15.5	A-	A-	3.667	A
111	Dawy Pillat	13	4.5	D+	B	1.667	C
112	Dean Roy	7	9	C+	B-	2.400	C
113	Desmond Hansie	4	8.5	C	A-	2.333	C
114	Gurpreet King	15	10.25	C+	C	2.267	C
115	Imrul Kahn	8	1.5	D-	B	1.133	D
116	Jaspreet Ahluwalia	15	12.5	B	C	2.800	B
117	Jitu Nagesh	2	6.75	C-	D	1.533	D
118	Kamaal Khanum	4	11.25	B-	A-	2.867	B
119	Kuni Senthil	1	7	C-	C+	1.800	C
120	Mani Panda	14	8.5	C	C	2.000	C
121	Mohsin Patel	13	14	B+	B	3.267	B
122	Nilu Phulket	15	0	F	C	0.400	F
123	Prashant Pramod	8	17	A	B	3.800	A
124	Preeti Singania	5	1.75	D-	B+	1.200	D
125	Priya Doshi	13	2.25	D-	B	1.133	D
126	Priyanka Sandy	11	12.75	B	C+	2.867	B
127	Rahul Gupt	7	11.75	B-	B-	2.667	B
128	Raja Chakrakant	6	9	C+	C-	2.200	C
129	Rajesh Channy	7	4.75	D+	B-	1.600	C
130	Ranbeer Shourya	4	4	D	A-	1.533	D
131	Ratul Kureshi	10	10.25	C+	D+	2.133	C

Exam Marks	Grade	Total Students(%)	GPA
0	F	4.88	0
1.5	D-	9.76	0.666667
3	D	9.76	1
4.5	D+	12.20	1.333333
6	C-	9.76	1.666666
7.5	C	7.32	2
9	C+	17.07	2.333333
10.5	B-	9.76	2.666666
12	B	7.32	3
13.5	B+	4.88	3.333333
15	A-	4.88	3.666666
16.5	A	2.44	4
18	A+	0.00	4.333333

Group No.	Grade
1	C+
2	D
3	A
4	A-
5	B+
6	C-
7	B-
8	B
9	A-
10	D+
11	C+
12	C+
13	B
14	C
15	C

CGPA	Letter Grade
0	F
0.55	D
1.55	C
2.55	B
3.55	A

Responses / Group Assignments / Grades / Student Database

Fig 4: Spreadsheet for Grading

The function "MCQ_evaluator" is defined (in the module of the existing workbook) as shown below:

```
Function MCQ_Evaluator (a As String, b As String, c As Single, d As Single) As Single
Dim score As Single
Dim i As Integer
For i = 1 To Len(a)
    If Mid(a, i, 1) = Mid(b, i, 1) Then
        score = score + c
    ElseIf Mid(a, i, 1) = "N" Then
        score = score
    Else
        score = score + d
    End If
Next i
MCQ_Evaluator = score
End Function
```

*2.2 Spreadsheet Model for Grading***

Once the evaluation of a given component like a quiz, a mid-term or an end-term is complete, the instructor assigns appropriate grades to the students for a given component based on their performance. Usually there is also a group component in the assessment scheme in which a student gets the same grade as his/her group members based on the overall performance of the group in the component. Once the evaluation for the various components (examinations, quizzes, group components, etc.) is complete, the final grade of a student for the course is computed based on the weightage given to individual components.

For the purpose of illustration (Fig 4), we assume there are only two components in the final evaluation of students: a.) exam (80%) and, b.) group component (20%). The grades given by an instructor for the two components range from F (fail) to A+ (excellent).The spreadsheet model for computing the grades for the

** Readers are suggested to refer to [6]-[9] for different spreadsheet models for automating the grading process. The model discussed in this note has been designed specifically keeping the context of WIMCI in mind.

same is described below:

1. An instructor can decide tentative grades for an individual component, as shown in the range K4:L16. The grades of individuals for that component are then computed using a VLOOKUP function. To check the percentage of students securing a certain grade, a combination of COUNTIF and COUNT functions are used. If the grade distribution for the component is not satisfactory, the ranges of the marks (K4:L16) are changed till the overall grade distribution for the component confirms to the institute's grading norms.
2. For assigning the group level grades to a student, a VLOOKUP function is used which retrieves the common grade corresponding to the group of the given student (range K19:L33).
3. The grades of various components, for an individual student, are converted into equivalent GPAs[4] (using N4:N16) and the final CGPA[5] (column G, row 4 onwards) is then computed by taking the weighted average of GPAs of all components.
4. The letter grade for a student for the course is then assigned based on his/her CGPA (N19:O24).

The actual formulae used in the sheet have been mentioned in Table 3.

Table 3: Formulae Entered in the Spreadsheet for Grading

S. No.	Formula	Entered in Cell(s)	Remark
1	=VLOOKUP(D4,$K$4:$L$16, 2,TRUE)	Enter in E4. To be copied in column E, row 4 onwards	Looks up the grade for the exam component in the grade table based on the marks scored by the student.
2	=VLOOKUP(C4,$K$19: $L$33,2,FALSE)	Enter in F4. To be copied in column F, row 4 onwards	Looks up the common grade corresponding to the group component based on the group number of the student.
3	=COUNTIF($E$4:$E$44,L4)* 100/COUNT($D$4:$D$44)	Enter in M4. To be copied in column M, row 4 onwards	Counts the percentage of students in the class getting a certain grade in the exam component.
4	=VLOOKUP(E4,$L$4:$N$16,3, FALSE)*$E$2+VLOOKUP (F4,$L$4:$N$16,3,FALSE)*$F$2	Enter in G4. To be copied in column G, row 4 onwards	Calculates the Cumulative Grade Point Average (CGPA) for a student by converting the grades obtained by a student in the exam and group components into equivalent Grade Point Average (GPA) and adding them.
5	=VLOOKUP(G4,$N$20: $O$24,2)	Enter in H4. To be copied in column H, row 4 onwards	Assigns a letter grade to the student in the course based on his/her CGPA.

4. Grade Point Average
5. Cumulative Grade Point Average

3. Results and Discussion

Once the spreadsheet model for evaluation and grading is ready, the actual effort required to evaluate and grade the students reduces significantly. As soon as the responses are typed, the spreadsheet displays the total marks for the given set of responses. Also, the effort required to type 50 responses is not too different from that required to type 20. If the students are asked to write their answers explicitly in a separate answer sheet instead of ticking the responses on the question paper itself, the time to read the responses from the question paper can be further saved. Since the model does not require manual crosschecking of the responses with the correct answer for the given set, there is no additional burden on the evaluator if the number of sets of the question paper is large. Also, a significant amount of time is saved in counting the correct and incorrect responses. In fact the time taken to type the roll number, the question set and the response string for all the students is the only time required for evaluation.

4. Conclusion

The model explained in the note may not be applicable in all situations but it can still be very useful in evaluating most of the MCQ type question papers, with few modifications. The model for grading is quite generic and it can still be implemented when one wants to calculate the final grade for a given course once the individual course components have been evaluated. The spreadsheet model is being successfully used at the institute by some of the faculty members and few academic associates for evaluating MCQ type examinations and overall course grading. The model has helped in significantly bringing down the time for evaluation and grading for a PGP class of 90 students thus speeding up the feedback process in a great manner.

References

1. Excel Help and How-to- Office.com (2013). Retrieved on January 23, 2013, from http://office.microsoft.com/en-us/excel-help/
2. Winston, W.L. (2011). Microsoft Excel 2010 Data Analysis and Business Modeling, 3rd edition, PHI Learning Private Limited- New Delhi
3. Use Google Forms to create a self-grading quiz. Retrieved on February 8, 2013, from http://www.techrepublic.com/blog/google-in-the-enterprise/use-google-forms-to-create-a-self-grading-quiz/1579
4. Google Apps, Module 4: Docs > Chapter 6: Forms. Retrieved on February 8, 2013, from http://edutraining.googleapps.com/Training-Home/module-4-docs/chapter-6/5-3
5. Using MS Excel to Analyze Data: A Tutorial. Retrieved on February 8, 2013, from http://www.sagepub.com/liustudy/chapters/08/UsingExceltoAnalyzeData.pdf
6. Letter Grades in Excel. Retrieved on February 8, 2013, from http://users.wfu.edu/matthews/plus_minus/excel_example.html
7. Excel 2010: Automating Grading Tasks. Retrieved on February 8, 2013, from http://ase.tufts.edu/its/trainDocuments/ excel10AutomaticGrading.pdf
8. Creating A Grade Sheet With Microsoft Excel. Retrieved on February 8, 2013, from http://www.math.ucla.edu/~tat/Docs/excel.pdf
9. Using an Excel Spreadsheet for Grades. Retrieved on February 8, 2013, from http://www.pcc.edu/resources/instructional-support/tools/hardware-software/documents/is-excel-for-grades.pdf

Harshal Lowalekar is an Assistant Professor in the Operations Management and Quantitative Techniques Area at Indian Institute of Management Indore. He is a Fellow of Indian Institute of Management Ahmedabad.