

## Assignment 2 (Group Assignment)

**Due: 23<sup>rd</sup> November 2022, 11:59 pm**

**(Warning: Peer evaluation will be conducted after the deadline. Students should make sure they make a fair contribution to the submitted assignment. An independent judgement is applied to review each case, and an appeal on the decision is **NOT** allowed.)**

### Group Forming

One member of your group will be responsible for assignment submission to Canvas. Students will work in a pre-assigned group.

### Learning Outcomes

1. Apply programming concepts to solve business problems
2. Describe the logic and flows of given programs
3. Predict the output of a program
4. Write programs with common programming practices
5. Identify and fix logical and runtime errors in programs

### Background

Grading is the last step for the completion of a university course.

To enhance the grading process, our university would like to develop an Excel application that can generate letter grades for students efficiently and effectively. This task is assigned to the ISOM department of the business school and project groups are formed to implement this application, under the supervision of Prof. James Kwok. ISOM3230 students are invited to participate in a part of the development of this project.

Currently our existing grade reports are generated manually with Microsoft Excel 2019 or Office 365. The school expects the newly developed VBA application that can be compatible with our existing practice.

Sample score sheets will be provided to project groups. Project groups are required to follow the project description to develop a VBA application for any score files with similar structure and format.

To illustrate how the VBA application works, some sample data and screenshots are shown below:

#### Before the execution:

	A	B	C	D	E	F	G	H	I
	Class Section	ID	Student Name	Quiz (15%)	Case (10%)	Midterm (35%)	Final (40%)	Total	TA Checking
1	L1	20103957	Eleni Parks	14.44	9.50	35.00	36.80	95.74	
2	L1	20104866	Nabeela Contreras	14.06	10.00	35.00	36.60	95.66	
3	L1	20105371	Joel Cornigan	13.88	9.50	33.60	37.60	94.58	
4	L1	20104967	Arlo Davies	15.00	9.50	33.60	36.20	94.30	
5	L1	20102240	Mohamad Stamp	15.00	9.50	33.60	35.60	93.70	
6	L1	20102745	Brian Huerta	14.25	10.00	32.20	35.80	92.25	
7	L1	20106482	Harlen Reader	14.81	10.00	32.20	34.60	91.61	
8	L1	20105270	Camel Merrill	11.25	9.50	32.20	37.20	90.15	
9	L1	20104058	Jaeden Curtis	12.41	9.00	30.80	37.60	89.81	
10	L1	20105068	Eryk Burris	15.00	9.50	28.00	37.20	89.70	
11	L1	20103755	Morris Leon	14.25	9.00	30.80	35.60	89.65	Checked
12	L1	20105876	Millie-Rose Avalos	15.00	8.50	29.40	35.80	88.70	
13	L1	20106179	Jemaine Bautista	14.44	9.00	30.80	34.00	88.24	
14	L1	20101937	Hugo Estrada	10.88	9.50	35.00	32.40	87.78	
15	L1	20100927	Angelika Portillo	13.28	9.00	29.40	36.00	87.68	
16	L1	20102846	Zishan Wynn	12.38	9.00	29.40	36.60	87.38	
17	L1	20105775	Alexa Chung	14.63	9.00	29.40	34.20	87.23	
18	L1	20105573	Nisha Gough	15.00	10.00	26.60	35.60	87.20	Checked
19	L1	20104159	Clement Wise	14.44	9.50	26.60	36.40	86.94	
20	L1	20106280	Natanial Blankenship	13.50	8.00	29.40	35.20	86.10	Checked
21	L1	20106583	Habibah Villalobos	12.38	9.50	30.80	33.20	85.88	
22	L1	20103654	Eshaan Garcia	15.00	10.00	25.20	35.60	85.80	
23	L1	20105674	Taine Rosas	13.88	9.50	26.60	35.80	85.78	
24	L1	20105169	Daanish Dotson	14.06	8.50	28.00	35.20	85.76	
25	L1	20106381	Codey Potts	14.06	9.50	29.40	32.60	85.56	
26	L1	20101533	Guto Ellison	12.94	9.50	29.40	33.20	85.04	
27	L1	20100220	Justine Paul	15.00	9.00	25.20	35.60	84.80	
28	L1	20103048	Evie-Mae Sparks	14.81	9.00	25.20	35.00	84.01	
29	L1	20101836	Juan Frazier	10.80	9.00	30.80	32.80	83.10	
30	L1	20101634	Danni Carpenter	14.44	10.00	26.60	31.20	82.24	

**After the execution:** The VBA application generates the following output based on the given dataset. (This below is just a sample output of the VBA application for your reference.)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
Class Section	Student Name	ID	Quiz (15%)	Case (10%)	Midterm (35%)	Final (40%)	Total	Difference	Grade	TA Checking	Total Student: 50					
L1	Elen Parks	20103957	14.44	9.50	35.00	36.80	95.74	0.08	A+							
L1	Nabeela Contreras	20104866	14.06	10.00	35.00	36.60	95.66	1.09	A+							
L1	Joel Corrigan	20105371	13.88	9.50	33.60	37.60	94.58	0.27	A+							
L1	Arlo Davies	20104967	15.00	9.50	33.60	36.20	94.30	0.60	A+							
L1	Mohamad Stamp	20102240	15.00	9.50	33.60	35.60	93.70	1.45	A+							
L1	Brian Huerta	20102745	14.25	10.00	32.20	35.80	92.25	0.60	A							
L1	Carmel Merrill	20105270	12.75	9.50	32.20	37.20	91.65	0.04	A							
L1	Harlen Reader	20106482	14.81	10.00	32.20	34.60	91.61	1.80	A							
L1	Jaeden Curtis	20104038	12.41	9.00	30.80	37.60	89.81	1.29	A-							
L1	Momis Leon	20103755	13.13	9.00	30.80	35.60	88.53	0.03	A-							
L1	Nisha Gough	20105573	13.50	10.00	29.40	35.60	88.50	0.26	A-							
L1	Jermaine Bautista	20106179	14.44	9.00	30.80	34.00	88.24	0.46	A-							
L1	Hugo Estrada	20101937	10.88	9.50	35.00	32.40	87.78	0.10	B+							
L1	Angelika Portillo	20100927	13.28	9.00	29.40	36.00	87.68	0.30	B+							
L1	Zishan Wynn	20102846	12.38	9.00	29.40	36.60	87.38	0.15	B+							
L1	Alexa Chung	20105775	14.63	9.00	29.40	34.20	87.23	0.03	B+							
L1	Millie-Rose Avalos	20105876	13.50	8.50	29.40	35.80	87.20	0.26	B+							
L1	Clement Wise	20104159	14.44	9.50	26.60	36.40	86.94	0.04	B+							
L1	Eryk Burns	20105068	15.00	9.50	25.20	37.20	86.90	0.80	B+							
L1	Nataniel Blankenship	20106280	13.50	8.00	29.40	35.20	86.10	0.22	B							
L1	Habbab Villalobos	20106583	12.38	9.50	30.80	33.20	85.88	0.07	B							
L1	Eshaan Garcia	20103654	15.00	10.00	25.20	35.60	85.80	0.03	B							
L1	Taine Rosas	20105674	13.88	9.50	26.60	35.80	85.78	0.01	B							
L1	Daanish Dotson	20105169	14.06	8.50	28.00	35.20	85.76	0.20	B							
L1	Codery Potts	20106381	14.06	9.50	29.40	32.60	85.56	0.24	B							
L1	Emmanuel Chan	20104462	13.13	9.00	29.40	33.80	85.33	0.29	B							
L1	Guto Elison	20101533	12.94	9.50	29.40	33.20	85.04	0.24	B							
L1	Justine Paul	20100220	15.00	9.00	25.20	35.60	84.80	0.79	B							
L1	Ernie-Mae Sparks	20103049	14.81	9.00	25.20	35.00	84.01	0.91	B-							
L1	Juan Frazier	20101836	10.50	9.00	30.80	32.80	83.10	0.89	C+							
L1	Fahad Sanderson	20101028	14.21	8.80	25.20	34.00	82.21	0.21	C+							
L1	Adem Velasquez	20104361	13.20	9.00	26.60	33.20	82.00	0.28	C+							
L1	Falak Ramsay	20102038	12.53	9.00	23.80	36.40	81.73	0.92	C+							
L1	Xavier Krause	20100826	12.60	9.00	26.60	32.60	80.80	0.70	C+							
L1	Kieron Hamington	20102644	12.90	9.00	28.00	30.20	80.10	0.66	C+							

### Project Description

There is no specific design of graphical user interface (GUI) for this assignment and students are required to come up with their own design. However, the VBA application needs to include the following basic functionalities, and the user interfaces need to effectively support them.

### Assumptions:

- The class size must be equal to or larger than 50 students
- The grade excel file may contain many columns, but the first three columns must be “Class Section”, “ID” and “Student Name”, the last two columns must also be “Total” and “TA checking” column.
- Course instructors will use MS Excel in virtual barn to run the application.
- Course instructors will use the VBA application by opening both the VBA application Excel file and the course grade file. Activate the course grade file, then generate grades for the course grade file by clicking the “Grade” button.

### Basic functionalities

#### 1. Userform

- When the excel file is launched, a userform should spontaneously be displayed with main buttons: (“Grade”, “Clear” and “Exit” – you may name and/or position the buttons in any sensible manner)

Basic Functionalities

Additional Functionalities

- Create a shortcut key “F4” to launch the grade system’s userform.

## 2. "Grade" button

When users click on the "Grade" button, the VBA application performs the functionalities, (i) to (v):

- (i) Sort Totals in descending order and all related data are updated accordingly.
- (ii) Create a "Difference" column
  - a. Create a new column right next to the "Total" column with the column heading "Difference"
  - b. Calculate the differences of the total scores (in "Total" column). The difference score of a student (in row 2) is the difference between the total score of the student and the total score of the student in row 3, i.e., H2-H3.
  - c. Display the difference in the "Difference" column.
- (iii) Create a "Grade" column
  - a. Create a new column right next to the "Difference" column with the column heading called "Grade"
  - b. Generate different grades and sub-grades according to guidelines and rules stated under "How to generate letter grades", by default, the grading will be based on **Instructor's preferences**.
  - c. Show the grades in the "Grade" column.
  - d. Highlight the cut off for each grade and sub-grade.
- (iv) Move the "TA Checking" column
  - a. This column is given in the original data file.
  - b. The VBA application moves the original column to the column next to "Grade".
- (v) Create a table of Summary
  - a. Summarize the grade results in the "Grade" column. Present the summary in a table format. See the screenshot below for the format and structure. All numbers in the table are generated and displayed by **Excel formula**. The summary needs to include:
    - (A) The total number of students (derived from Column B)
    - (B) The list of sub-grades
    - (C) The number of students who get that particular sub-grade
    - (D) The percentage of students who get that particular sub-grade
    - (E) The percentage of students in that grade range
    - (F) The total number of students (derived from the Summary table)
    - (G) The total number of students (derived from (C))

Total Student:	19			
A+	2	10.53%		
A	1	5.26%		26.32%
A-	2	10.53%		
B+	2	10.53%		
B	3	15.79%		31.58%
B-	1	5.26%		
C+	3	15.79%		
C	2	10.53%		31.58%
C-	1	5.26%		
D	1	5.26%		5.26%
F	1	5.26%		5.26%
	19			100.00%

Callouts in the image:

- (A) points to the total student count (19).
- (B) points to the list of sub-grades (A+, A, A-, B+, B, B-, C+, C, C-, D, F).
- (C) points to the number of students for a specific sub-grade (3 for B).
- (D) points to the percentage of students for a specific sub-grade (15.79% for B).
- (E) points to the percentage of students in a grade range (31.58% for B).
- (F) points to the total number of students derived from the summary table (19).
- (G) points to the total number of students derived from (C) (3 for B).

## 3. "Clear" button

When users click on the "Clear" button, the VBA application deletes ALL data created and generated by this application and restore the Excel file to its original status (Including the order of the data).

### Additional functionality (as Creativity)

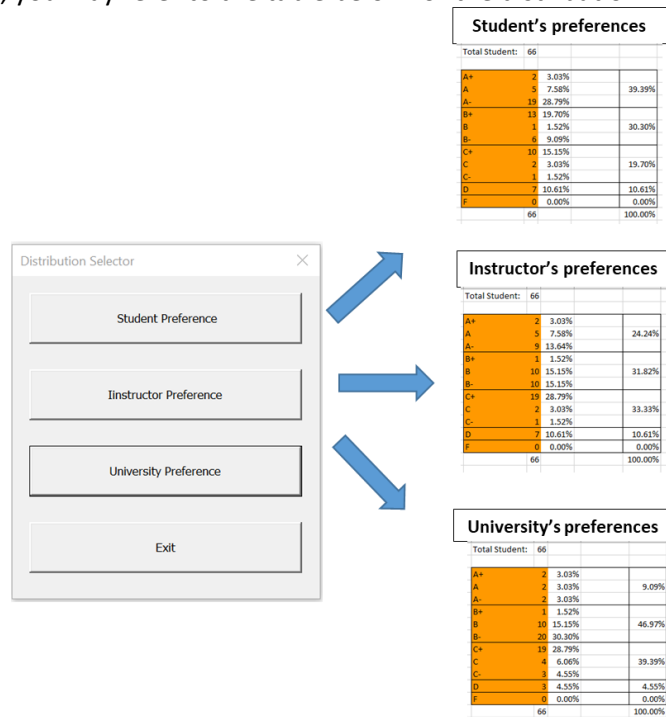
In addition to the basic functions, the VBA application is required to provide at least **THREE** additional functions. This is regarded as the “creativity” part of this assignment as students may choose what to implement. So, students need to propose and implement some practical features for the grading application. Below are some recommendations.

- Add an additional program to generate a separate Excel file that contains the following columns and the summary table. The columns and the summary are directly copied from the grade sheet, and the summary contains **plain text ONLY (No formula should be included)**.

	A	B	C	D	E	F	G	H	I
1	<b>ID</b>	<b>Student Name</b>	<b>Grade</b>						
2	20100119	Coby White	A+						
3	20100220	Justine Paul	A+		Total Student:	66			
4	20100321	Sanaya Downs	A						
5	20100422	Jack Beasley	A		A+	2	3.03%		
6	20100523	Zachery Quintero	A		A	5	7.58%		24.24%
7	20100624	Amara Maldonado	A		A-	9	13.64%		
8	20100725	Julia Ratliff	A		B+	1	1.52%		
9	20100826	Xavier Krause	A-		B	10	15.15%		31.82%
10	20100927	Angelika Portillo	A-		B-	10	15.15%		
11	20101028	Fahad Sanderson	A-		C+	19	28.79%		
12	20101129	Tianna Grey	A-		C	2	3.03%		33.33%
13	20101230	Steven Liu	A-		C-	1	1.52%		
14	20101331	Areebah Petersen	A-		D	7	10.61%		10.61%
15	20101432	Yuvraj Lees	A-		F	0	0.00%		0.00%
16	20101533	Guto Ellison	A-			66			100.00%
17	20101634	Danni Carpenter	A-						
18	20101735	Juliette Seymour	B+						

- Add an additional Userform to provide more grade distribution options. The grade distribution option userform may pop up after users have clicked the “Grade” button. But it is assumed that the grading function should have been performed before providing grade distribution options.

Here are some examples of grade distribution option. The userform has 3 different grade distribution options. For student’s preference, you need to get the distribution from the user by userform. For the rest, you may refer to the table below for the distribution.



- Add an additional function to validate the correctness of the data by comparing the student scores in assessment worksheets, e.g., “Quiz”, “Case” worksheets and the student scores in “Main” worksheet. If the data is correct, highlight the cells with one color. If the data is incorrect, highlight the data cells with different colors and provide a warning message.

### How to generate letter grades

- There are 3 possible grading distributions in this application, namely **student’s preferences**, **instructor’s preferences**, and **university’s preferences**.
- Users may specify the percentages of Grades A, B, and C only, while grades D and F are determined by the application.
- The sum of the percentages of A, B, C cannot be more than 100%.
- The percentages in the instructor’s preferences are fixed and it is a default option.
- The percentages in the university’s preferences are fixed and cannot be changed.
- The percentages in the student’s preferences can be changed to any “reasonable” and positive values.

Here are the steps to generate letter grades for students

**Step 1: Let users choose a grade distribution option and use the default values (or set the values)**

	A	B	C	D	F
Student’s preferences	$\leq 40\%$ and $\geq 5\%$	$\leq 40\%$ and $\geq 5\%$	$\leq 20\%$ and $\geq 5\%$	-	-
Instructor’s preferences (default setting)	$= 25\%$	$= 32\%$	$= 34\%$	-	-
University’s preferences	$= 10\%$	$= 50\%$	$= 40\%$	-	-

(Assume that the instructor’s preferences are chosen.)

**Step 2:**

(a) Calculate the total numbers of students per grade. (Noted: Round off the number if it is not an integer)

	A	B	C	D	F
Instructor’s preferences (default setting)	$100 \times 25\% = 25$	$100 \times 32\% = 32$	$100 \times 34\% = 34$	-	-

(Assume that there are 100 students in the class.)

(b) The cut-off of grade F solely depends on the differences of the last 3 students.

If the difference between the last and the last second one is so huge, such as 15%. The last student will receive a F.

If the difference between the last second and the last third is so huge, such as 15%. Both the last and the last second students will receive a F.

The below is an example.

Total	Difference	Grade
72.19	0.38	
71.81	0.20	
71.61	3.60	
68.01	34.01	
34.00		F

Here are the updated percentages.

	A	B	C	D	F (finalized)
Instructor's preferences (default setting)	$100 \times 25\% = 25$	$100 \times 32\% = 32$	$100 \times 34\% = 34$		<b>1</b> (= 1%)

(c) Find the best cut-offs.

Example: Find the best cut-offs for grades A and B.

- Check the TWO adjacent Differences, i.e., +2 and -2 differences (highlighted in yellow below). So the application needs to consider 5 numbers in total.

Difference	Grade
0.30	A
0.04	A
0.11	A
0.05	A
0.95	B
0.46	B
0.46	B

25th

- Find the highest one. In this case, it is 0.95. Therefore, the cutoff should be at the 26<sup>th</sup> student as follows. And the percentage of A becomes 26%.

Difference	Grade
0.30	A
0.04	A
0.11	A
0.05	A
0.95	A
0.46	B
0.46	B

26th

Here are the updated percentages.

	A (finalized)	B	C	D	F (finalized)
Instructor's preferences (default setting)	<b>26</b> (= 26%)	$100 \times 32\% = 32$	$100 \times 34\% = 34$		<b>1</b> (= 1%)

Repeat the above steps to find the best cut-offs for (i) grades B and C, and (ii) grades C and D.

Example: Find the best cut-offs for grades B and C. Since the percentage of grade A has been changed to 26%, the application will locate the 58<sup>th</sup> student. Then find the highest difference within those 5 numbers.

Difference	Grade
0.09	B
0.53	B
0.20	B
0.04	B
0.79	C
0.91	C
0.86	C

58th

And the final decision is as follows.

	Difference	Grade
	0.09	B
	0.53	B
	0.20	B
	0.04	B
	0.79	B
60th	0.91	B
	0.86	C
	0.03	C

Here are the updated percentages.

	A (finalized)	B (finalized)	C	D	F (finalized)
Instructor's preferences (default setting)	26 ( = 26%)	34 ( = 34%)	100 x 34% = 34		1 ( = 1%)

The next step is to find the cut-off of grades C and D. The application needs to locate the 94th (26+34+34=94) student.

	Difference	Grade
	0.40	C
	1.30	C
	0.08	C
94th	0.80	C
	0.15	D
	0.95	D
	0.51	D

Finally, the cut-off is at the 92nd student.

	Difference	Grade
	0.40	C
92th	1.30	C
	0.08	D
	0.80	D
	0.15	D
	0.95	D
	0.51	D

The final percentages are as follows.

	A (finalized)	B (finalized)	C (finalized)	D	F (finalized)
Instructor's preferences (default setting)	26 ( = 26%)	34 ( = 34%)	32 ( = 32%)	100-26-34-32-1 ( = 7%)	1 ( = 1%)

**Step 3: Determine the percentages of sub grades A, B, and C only (Grades D and F do not have sub-grades). The task is to find the BEST cut-offs for the sub-grades. If one of the cut-offs is also the cut-off for letter grade, choose the third one.**

**Based on the results in Step 2, there are 26 students in A range. Here are two common situations: (a) normal case, and (b) a case with more than two same values.**

Normal case			4 same scores		
Original	Sorted	Sub grades	Original	Sorted	Sub grades
No. Difference	No. Difference	No. Difference	No. Difference	No. Difference	No. Difference
1 0.08	7 1.46	1 0.08 A+	1 0.08	7 1.46	1 0.08 A+
2 1.09	5 1.45	2 1.09 A+	2 1.09	12 1.46	2 1.09 A+
3 0.27	2 1.09	3 0.27 A+	3 0.27	17 1.46	3 0.27 A+
4 0.60	11 0.95	4 0.60 A+	4 0.60	21 1.46	4 0.60 A+
5 1.45	26 0.95	5 1.45 A+	5 1.45	5 1.45	5 1.45 A+
6 0.64	19 0.84	6 0.64 A	6 0.64	2 1.09	6 0.64 A+
7 1.46	6 0.64	7 1.46 A	7 1.46	11 0.95	7 1.46 A+
8 0.34	4 0.60	8 0.34 A-	8 0.34	26 0.95	8 0.34 A
9 0.11	12 0.46	9 0.11 A-	9 0.11	19 0.84	9 0.11 A
10 0.05	13 0.46	10 0.05 A-	10 0.05	6 0.64	10 0.05 A
11 0.95	8 0.34	11 0.95 A-	11 0.95	4 0.60	11 0.95 A
12 0.46	22 0.30	12 0.46 A-	12 1.46	13 0.46	12 1.46 A
13 0.46	15 0.30	13 0.46 A-	13 0.46	8 0.34	13 0.46 A-
14 0.10	3 0.27	14 0.10 A-	14 0.10	22 0.30	14 0.10 A-
15 0.30	18 0.26	15 0.30 A-	15 0.30	15 0.30	15 0.30 A-
16 0.15	20 0.22	16 0.15 A-	16 0.15	3 0.27	16 0.15 A-
17 0.02	16 0.15	17 0.02 A-	17 1.46	18 0.26	17 1.46 A-
18 0.26	9 0.11	18 0.26 A-	18 0.26	20 0.22	18 0.26 A-
19 0.84	24 0.11	19 0.84 A-	19 0.84	16 0.15	19 0.84 A-
20 0.22	14 0.10	20 0.22 A-	20 0.22	9 0.11	20 0.22 A-
21 0.07	1 0.08	21 0.07 A-	21 1.46	24 0.11	21 1.46 A-
22 0.30	21 0.07	22 0.30 A-	22 0.30	14 0.10	22 0.30 A-
23 0.04	25 0.05	23 0.04 A-	23 0.04	1 0.08	23 0.04 A-
24 0.11	10 0.05	24 0.11 A-	24 0.11	25 0.05	24 0.11 A-
25 0.05	23 0.04	25 0.05 A-	25 0.05	10 0.05	25 0.05 A-
26 0.95	17 0.02	26 0.95 A-	26 0.95	23 0.04	26 0.95 A-

### Video presentation

Student groups are required to produce a video presentation file (in mp4 format). The duration should not be more than 10 minutes. Here are the topics to be included in the presentation.

- Introduction (objectives)
- Overview of the program
- Demo
- Highlights of the program (explain how to solve critical parts of your program)
- Problems encountered
- Further improvements and future works

All students in a group need to be involved in the presentation. They are required to present their own part(s) of the program.

### Marking Criteria

Area
<b>Grading Main Functions</b> <b>(Relevant columns, names, calculations)</b> The VBA application needs to <ul style="list-style-type: none"> <li>• Use a worksheet called “Main” to record the overall grade information (different assessments, differences, grades etc.)</li> <li>• Implement all basic requirements</li> <li>• Be in comply with stated “assumptions”</li> </ul> All functionalities must be presented in Methods (either Sub or Function) with meaningful names
<b>Additional functionality (Creativity)</b> <ul style="list-style-type: none"> <li>• The VBA application needs to implement at least 3 additional functionalities</li> <li>• All additional functions must be presented in Methods (either Sub or Function) with meaningful names</li> <li>• These additional functions have to be practical and useful</li> <li>• These additional functions need to be reasonably complex</li> </ul>
<b>GUI (Creativity)</b> <ul style="list-style-type: none"> <li>• Use userform effectively</li> </ul>



<ul style="list-style-type: none"> <li>• Enable the shortcut key “F4” to activate the application</li> <li>• The VBA application needs to be user friendly.</li> </ul>
<b>Programming Styles</b> <ul style="list-style-type: none"> <li>• Best use of VBA programming language e.g., the VBA application needs to use variables, objects, array, control flow, functions and subroutines properly</li> <li>• Best programming practice e.g., The code is well organized and with proper and consistent indentation</li> </ul>
<b>Comments</b> <ul style="list-style-type: none"> <li>• The VBA application must contain comments to explain the codes. All the comments must be relevant.</li> </ul>
<b>Video presentation</b> <ul style="list-style-type: none"> <li>• Video quality, including audio and video</li> <li>• Presentation skills</li> </ul>

## FAQs

### 1. Are we allowed to use built-in function for this program?

For this program, you are allowed to use built-in functions provided that the function itself is generated from the macro instead of typing it in the cells manually.

Some examples of built-in function that you will find useful for this assignment:

(Note these are just for your reference, you must modify the variables and the parameters may not be accurate)

=COUNTIF(range, criteria)

=SUM(numbers)

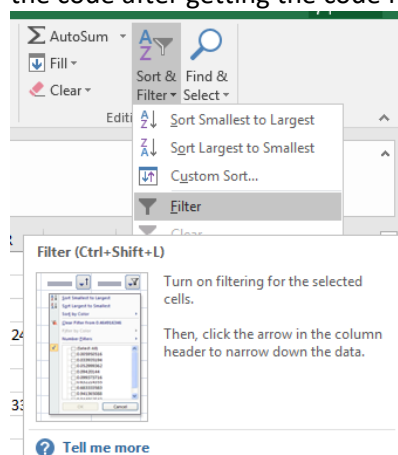
=VLOOKUP(value, array, return value, true or false))

### 2. What if I don't have userform in my computer because I am using Mac, will I get marks deducted?

At the start of the course, we recommend students who use Mac to install virtual barn or bootcamp to solve the incompatibility issue. We expect students to have them solved by now. We recommend students to work on the basic codes on their own machines first, and work with a different machine or virtual barn for userform to minimize the use of virtual barn due to possible delay with virtual barn.

### 3. I do not know how to filter the data into descending order, how can I do it with VBA?

You may use the built-in tools on the tool bar to do so with macro recorder. The tool is under “Files” -> “Sort & Filter” -> “Filter”. Please remember to highlight the relevant dataset to do the filtering. Modify the code after getting the code from macro recorder.



### *Submission Guidelines*

1. All the grading will be done by using Microsoft Excel in Virtual Barn. Please ensure your program can work in the Excel program in Virtual Barn.
2. Please zip your programming files and name the zip file as your Group such as Group1.zip.
3. Submit your zip file to Canvas.
4. Multiple submissions are allowed but only the last submission will be graded. Therefore, you need to **make sure the last submission is the best, and more importantly, it works.**
5. Late submission within 24 hours will result in a penalty of 30% deduction in your total marks. No further late submission is allowed.
6. Plagiarism (Copying) is a serious concern. The **MINIMUM** penalty is zero marks.  
Here are the rules to follow.

- Allow to use outside codes (not another student's/group's codes) but **must cite the sources of outside codes as comments.**
- We will use software to evaluate program similarity. When similar programs are found, the involved students/groups will get zero marks.
- Follow the section on "Academic honesty" in the course syllabus.