DEPARTMENT OF ECONOMICS
AND BUSINESS ECONOMICS
AARHUS UNIVERSITY

EXAM ASSIGNMENT

Study programme and level	**MSc Logistics and Supply Chain Management + elective**						
Term	V20-21o						
Course name and exam code(s)	Tools for Analytics						460202E016
Exam form and duration	WHAI (changed from WOAI due to Covid restrictions)						6 hours
Date and time	21 December, 2020						9.00 – 15.00
Supplementary material/aids	All	X	Specified		No		
Hand-in of hand-written material allowed	Yes		No	X	Comments:		
Anonymous exam?	Yes	X	No		Comments: Please do not write your name or student ID number anywhere.		
Number of pages (incl. front page)	9						

How to hand in your exam paper
Start preparing the hand in well in advance of the exam deadline.
Your exam paper should be handed in as a set of files (.Rmd, .html and .xlsm) under **'appendix material'**. The total maximum permitted file size here is 5 GB. Due to the system, you must also upload an empty pdf-document named after yourFlowId.pdf (max 200 MB).

If you experience problems uploading and handing in your exam paper in WISEflow, you can send the paper to the following email address: bss.exam@au.dk. You need to ask for permission to hand your paper for final assessment in WISEflow. Use the formula "Exemption" under "Applications to Study Councils" in the Student Self-Service. You need to apply as soon as possible after sending your paper to the email address.
If you need technical assistance during the exam, you can contact BSS IT-support, phone: 8715 0933. Contact the invigilator, if the exam is on-site.

Be aware that exam papers are as a rule only permitted for final assessment, if handed in in the right format/size and within the exam deadline.

Avoid being suspected of exam cheating

Remember to use quotation marks and to insert references if you copy text from other sources, incl. indicative exam solutions (plagiarism) or if you re-use parts of a previously submitted (passed) exam paper (self-plagiarism). Do not share your exam paper with others, or communicate about the assignment during the exam. Students must answer the exam assignment individually.
All submitted exam papers will be checked for plagiarism, so exam cheating (incl. collaboration between students) will be detected.

Questions during the first hour of the exam:

In case you have any *clarification* questions during the first hour of the exam, please email them to Lars Relund at larsrn@econ.au.dk (R) or Sanne Wøhlk at sanw@econ.au.dk (VBA). Do not expect instant answers. Answers to questions of a general concern will be posted to Blackboard.

Practical information:

- This exam is open book, open internet, closed other people. You may use any online or book-based resource you would like, but you must include citations for any code that you use (directly or indirectly). You **may not** consult with anyone else during this exam. You cannot ask direct questions on the internet, or consult with each other, not even hypothetical questions.
- This assignment has one appendix available for download from WISEflow. The file is a zip file containing the files you may need during the exam.
- The exam has a VBA and R part with approx. equal weight.
- Please note that the weights on each assignment are only guideline weights, and they only provide information regarding the relative weight of the assignments. The final evaluation will be given based on the total material handed in.
- If you find that some information is missing in the assignments, you may make the necessary assumptions and clearly specify these.
- Handing in: You must hand in a set of files (.Rmd, .xlsm and .html) as "Appendix material". Due to the system, you must also upload an empty pdf-document named `yourFlowId.pdf`.
- Your VBA code will be tested using Excel 2016 and your R code will be tested using R 4.0.3. As operating system Windows will be used.
- **About assignments 1-3: R**
    - Your R code must be written up in an R Markdown (Rmd) file named `yourFlowId.Rmd`. Moreover, also hand in the rendered/knitted html file.
    - Your file must include your code and a (brief) comment for each question. For example, "The three companies with smallest profit are …" or "The plot shows that …".
    - An R markdown template file is given in the appendix that you may use as a starting point.
    - You may load and use the following packages:
        - `library(tidyverse)`
        - `library(skimr)`
- **About assignments 4-6: VBA**
    - Your VBA code should be contained in a single Excel file named `yourFlowId.xlsm`.
    - Do not protect your code with password or turn it into an Add-In.

Assignment 1 - R (12%)

The euclidean distance between two points $p = (p_1, p_2)$ and $q = (q_1, q_2)$ can be calculated using formula

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}.$$

Question 1

Calculate the distance between points $p = (10{,}10)$ and $q = (4{,}3)$ using the formula.

Question 2

Consider 4 points in a matrix (one in each row):

```
p_mat <- matrix(c(0, 7, 8, 2, 10, 16, 8, 12), nrow = 4)
p_mat
```

```
##      [,1] [,2]
## [1,]    0   10
## [2,]    7   16
## [3,]    8    8
## [4,]    2   12
```

The distance matrix of `p_mat` is a 4 times 4 matrix where entry (i,j) contains the distance from the point in row `i` to the point in row `j`. Calculate the distance matrix of `p_mat`.

Question 3

Create a function `calc_distances` with the following features (implement as many as you can):

- Takes a matrix `p_mat` with a point in each row as input argument.
- Takes two additional input arguments `from` and `to` with default values `1:nrow(p_mat)`
- Return the distance matrix with values calculated for rows in the `from` input argument and columns in the `to` input argument. The other entries equals `NA`.
- The function should work for different `p_mat` (you may assume that the matrix always has two columns).

You may test your code using:

```
p_mat <- matrix(c(10, 9, 15, 15, 11, 19, 12, 11, 7, 15), nrow = 5)
calc_distances(p_mat)
calc_distances(p_mat, to = 3:4)
calc_distances(p_mat, from = c(1, nrow(p_mat)), to = 3:4)
```

Assignment 2 - R (22%)

The dataset `companies`, given in the appendix, lists approx. 1000 of the world's biggest companies, measured by sales, profits, assets and market value. The column/variables are:

- `name`: the name of the company.
- `country`: the country the company is situated in.
- `category`: the products the company produces.
- `sales`: the amount of sales of the company in billion USD.
- `profits`: the profit of the company in billion USD.
- `assets`: the assets of the company in billion USD.
- `marketvalue`: the market value of the company in billion USD.

Use the *dplyr* package in *tidyverse* to calculate relevant summary tables (data frames) and answer the following questions.

Question 1

Read the dataset file `companies.csv` into the dataset `companies`.

Question 2

How many different companies are we considering? How many different product categories and how many different countries? Hint: the *skimr* package might be useful.

Question 3

What are the three biggest companies with respect to market value?

Question 4

For each country, find the company with highest profit. What company has the highest profit in Denmark?

Question 5

Which four product categories have the highest total market value?

Question 6

Create a new data frame only containing rows from Denmark and with columns `name`, `category` and a column `value`, which equals the sum of columns `profits`, `assets` and `marketvalue`. Which company have the lowest value?

Assignment 3 - R (16%)

Answer this assignment using the *ggplot2* package in *tidyverse* (you might need the dplyr package for preparing the datasets you want to plot). We work with the dataset `companies` from Assignment 2 which can be read using:

```
companies <- read_csv("companies.csv")
```

Question 1

Create a visualization showing the number of companies for each product category with the following features:

- Number of companies is represented using bars and sorted increasingly or decreasingly.
- Informative figure title and axis titles are given.
- The labels on the x-axis are rotated 90 degrees.

What product category has the lowest number of companies?

Question 2

Consider product categories `Drugs & biotechnology` and `Media`. Create a visualization showing the profit given sales of each company with the following features:

- Different colours are used for each product category.
- Informative figure title and axis titles are given.
- A trend line for each category is added using `geom_smooth`.

Based on the trend lines, which product category gives the best profit?

Question 3

Consider product categories `Banking` and `Aerospace & defense`. Let `ratio` denote a variable/column that equals profit divided by sales. Create a visualization showing the variation in `ratio` with the following features:

- Different colours are used for each product category.
- Informative figure title and axis titles are given.

Based on the visualization comment on the variation and median. Which product category gives the highest ratio?

Question 4

The `continents` dataset given in the appendix matches countries to continents and contains two columns:

- `country`: the country.
- `continent`: the corresponding continent.

You can load the dataset using:

```
continents <- read_csv("continents.csv")
```

Consider product categories `Banking`, `Aerospace & defense`, `Telecommunications services` and `Semiconductors`. Create a visualization showing assets given market value for each company with the following features (hint: you may need to do a mutating join):

- Two continents `Americas` and `Europe` are considered.
- Different colours are used for each continent.
- A plot is given for each product category (facet).
- Informative figure title and axis titles are given.
- A trend line for each category is added using `geom_smooth(method = lm)`.

Based on the visualization, consider the trend lines for *Banking* and comment on these.

Assignment 4 - VBA (10%)

Please consult "Sheet4" in the Excel file "Assignments4-6.xlsx" that can be downloaded from WISEflow, and put the statement "Worksheets("Sheet4").Activate" in the top of your code to ensure that the code is using the correct sheet.

In Sheet4, you see seven data sets. Each data set consists of a list of integer values and is contained in a single column. Your code should be able to run on any of these data sets, but only on one data set at a time. The value in cell C1 states the column to use, so you can change the data set by changing this value (the values can be 1, 3, 5,…).

The data sets vary in size. If you need to know the number of values in the data set, it should be done as part of your vba code.

Write a sub, "*Assignment4*", that first stores the values of the data set indicated in cell C1 in an array, and then creates a matrix, "*Equal*", where the *Equal(k,j)* is 1, if the *k*'th and *j*'th values are equal, and 0 otherwise.

Assignment 5 - VBA (10%)

Create a sub, "*Assignment5*" that does the following:

1. Creates a new worksheet and names it "Sheet5 - X", where "X" is replaced by either the current date or the current date and time.
2. Uses one or several input boxes to obtain three integer numbers to be used in a RGB-colour code. You may assume that the user either does not enter anything or enters an

integer number (i.e. you may assume that the user does not enter e.g. text or floating point numbers), but you may not make assumptions regarding the size of the integer numbers provided. Your code must be able to handle this, such that eventually you obtain three numbers that are useful for RGB-colour coding (recall that they must be between 0 and 255).

3. Uses the three numbers obtained in (2) to change the colour of cell A1 of the sheet created in (1).

Assignment 6 - VBA (30%)

Please consult "Sheet6" in the Excel file and put the statement "Worksheets("Sheet6").Activate" in the top of your code to ensure that the code is using the correct sheet.

Consider a case with a virus that has infected a number of persons. A possible cure has been developed, but the effect of it is expected to be dependent on the persons' height. The cure can be tested on non-infected persons and the findings of this test can be directly transferred to any infected person whose height is within a range of 2 cm from the height of the tested person. For example, if the cure is tested on a non-infected person of height 172.2, then any infected person whose height is in the interval [170.2 ; 174.2] is *covered* by the test.

	A	B	C	D	E	F	G	H
1	Number of infected persons			30				
2	Number of test volunteers			10				
3	Number of volunteers to select			3				
4								
5	Infected persons					Test volunteers		
6	Person ID	Hight (cm)	Covered			Person ID	Hight (cm)	Selected
7	1	200,87				1	159,14	
8	2	196,43				2	197,21	
9	3	159,31				3	201,17	
10	4	204,5				4	202,96	
11	5	160,25				5	158,66	

Figure 1. Illustration of data for assignment 6.

Figure 1 shows the first 11 rows of sheet6. Cell D1 states the number of infected persons (n), and for each of them, columns A and B provide the person's ID and his/hers height, respectively. Cell D2 states the number of non-infected persons (m) volunteering to be test persons (we call them test volunteers), and for each of them, columns F and G provide their ID and height, respectively. The testing process is extremely resource demanding, and thus it is only possible to test a limited number of test volunteers (k). This number is stated in cell D3.

The purpose of this assignment is to write a sub, "*Assignment6*" that can be used to select the (k) test volunteers among the (m) available test volunteers to be used for tests and to determine which of the (n) infected persons are covered by the performed tests (the concept of covering is described above).

The selection problem is difficult to solve, so your code should use the following greedy strategy:

- The first test volunteer is selected as the one that can cover most infected persons (if several test volunteers can cover the same number of infected persons, select the one with the smallest ID).
- Continue until a total of k test volunteers has been selected by repeatedly selecting the test volunteer that can cover most infected persons not yet covered (if several test volunteers can cover the same number of uncovered infected persons, select the one with the smallest ID). Of course, the same test volunteer should not be selected twice.

Your code should generate the following output in columns C and H: In column H, your code should write a '1' for the selected test volunteers, and in columns C, your code should write a '1', if the infected person is covered.