ATIVIDADE PRÁTICA

MAP2112 – Introdução a Lógica de Programação e Modelagem Computacional

Prof. Dr. Luis Carlos de Castro Santos **1º semestre 2019**

Os exercícios abaixo são selecionados de:

https://www.r-exercises.com/start-here-to-learn-r/

o site contém mais de 4000 exercícios e serve como mais um auxílio ao aprendizado. Alguns comandos podem não ter sido apresentados durante as aulas, aproveite a oportunidade para conhecê-los use o help do R Studio. O site também contém as respostas aos exercícios, mas procure não utilizá-las, tente encontrar a sua solução mesmo que por tentativa e erro. Lembre-se que pode haver mais de uma forma de se obter o mesmo resultado e todas são válidas.

Para cada questão procure usar a cola fornecida com o material da aula 04 e depois execute o código em R para verificar sua resposta.

VECTOR EXERCISES 1:

Exercise 1

Consider two vectors, x, y

x=c(4,6,5,7,10,9,4,15)

y=c(0,10,1,8,2,3,4,1)

What is the value of: x*y

Exercise 2

Consider two vectors, a, b

a=c(1,2,4,5,6)

b=c(3,2,4,1,9)

What is the value of: cbind(a,b)

Exercise 3

Consider two vectors, a, b

a=c(1,5,4,3,6)

b=c(3,5,2,1,9)

What is the value of: a<=b

Exercise 4

Consider two vectors, a, b

a=c(10,2,4,15)

b=c(3,12,4,11)

What is the value of: rbind(a,b)

Exercise 5

If x=c(1:12)

What is the value of: dim(x)

What is the value of: length(x)

Exercise 6

If a=c(12:5)

What is the value of: is.numeric(a)

Exercise 7

Consider two vectors, x, y

x=c(12:4)

y=c(0,1,2,0,1,2,0,1,2)

What is the value of: which(!is.finite(x/y))

Exercise 8

Consider two vectors, x, y

x=letters[1:10]

y=letters[15:24]

What is the value of: x<y

Exercise 9

If x=c('blue','red','green','yellow')

What is the value of: is.character(x)

Exercise 10

If x=c('blue',10,'green',20)

What is the value of: is.character(x)

VECTOR EXERCISES 2:

Exercise 1

Consider a vector:

x <- c(4,6,5,7,10,9,4,15)

What is the value of:

c(4,6,5,7,10,9,4,15) < 7

a. TRUE, FALSE, TRUE, FALSE, FALSE, FALSE, TRUE, FALSE

b. TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, TRUE, FALSE

c. FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, TRUE, FALSE

d. TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, FALSE

e. TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, TRUE, FALSE

Exercise 2

Consider two vectors:

p <- c (3, 5, 6, 8)

and

q <- c (3, 3, 3)

What is the value of:

p+q

a. 6, 8, 6, 8

b. 6, 8, 0, 0

c. 6, 8, NA, NA

d. 3, 5, 6, 8

Warning message: In p+q : longer object length is not a multiple of shorter object length

e. 6, 8, 9, 11

Exercise 3

If:

Age <- c(22, 25, 18, 20)

Name <- c("James", "Mathew", "Olivia", "Stella")

Gender <- c("M", "M", "F", "F")

then what is the R-code for getting the following output;

Age Name Gender

1 22 James M

2 25 Mathew M

a.

DataFrame = data.frame(c(Age), c(Name), c(Gender))

subset(DataFrame, Gender == "M")

b.

DataFrame = data.frame(c(Age),c(Name),c(Gender))

subset(Gender=="M"), eval=FALSE

c.

DataFrame = data.frame(Age,Name,Gender)

subset(DataFrame,Gender=="M")

d.

DataFrame = data.frame(c(Age,Name,Gender))

subset(DataFrame,Gender=="M")

Exercise 4

If

z <- 0:9

then what is the output from the following R-statements:

digits <- as.character(z)

as.integer(digits)

a. Error in subset. object 'z' not found

b. 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

c. "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA"

d. "0", "1", "2", "3", "4", "5", "6", "7", "8", "9"

e. 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Exercise 5

Consider the vector:

x <- c(1,2,3,4)

What is the value of k for:

(x+2)[(!is.na(x)) & x > 0] -> k

a. 1, 2, 3, 4

b. 1, 4, 9, 16

c. Error: object 'k' not found

d. 3, 4, 5, 6

e. numeric(0)

Exercise 6

Consider the AirPassenger data set

data(AirPassengers)

Which statement will produce the following output?

[1] 112 118 132 129 121 135 148 148 136 119 104 118

a. AirPassengers[time(AirPassengers) >= 1949 & time(AirPassengers) < 1950, 12]

b. AirPassengers[AirPassengers >= 1949 & AirPassengers < 1950]

c. AirPassengers[time(AirPassengers) >= 1949 & time(AirPassengers) < 1950]

d. AirPassengers[AirPassengers >= 1949 & AirPassengers < 1950, 12]

e. c[[1]]

Exercise 7

If

x <- c(2, 4, 6, 8)

and

y <- c(TRUE, TRUE, FALSE, TRUE)

What is the value of:

sum(x[y])

a. 20

b. 8

c. 14

d. NA

Exercise 8

Consider the vector:

x <- c(34, 56, 55, 87, NA, 4, 77, NA, 21, NA, 39)

Which R-statement will count the number of NA values in x?

a. count(is.na(X))

b. length(is.na(x))

c. sum(is.na(x))

d. count(!is.na(x))

e. sum(!is.na(x))

LOGICAL VECTORS AND OPERATORS

Before you start, enter the following code:

data <- mtcars

Exercise 1

Use logical operators to output only those rows of data where column mpg is between 15 and 20 (excluding 15 and 20).

Exercise 2

Use logical operators to output only those rows of data where column cyl is equal to 6 and column am is not 0.

Exercise 3

Use logical operators to output only those rows of data where column gear or carb has the value 4.

Exercise 4

Use logical operators to output only the even rows of data.

Exercise 5

Use logical operators and change every fourth element in column mpg to 0.

Exercise 6

Output only those rows of data where columns vs and am have the same value 1, solve this without using == operator.

Exercise 7

(TRUE + TRUE) * FALSE , what does this expression evaluate to and why?

Exercise 8

Output only those rows of data where at least vs or am have the value 1, solve this without using == or !=.

Exercise 9

Explain the difference between | , || , & and &&.

Exercise 10

Change all values that are 0 in the column am in data to 2.

Exercise 11

Add 2 to every element in the column vs without using numbers.

Exercise 12

Output only those rows of data where vs and am have different values, solve this without using == or !=

MISSING VALUES

Exercise 1

If X <- c (22,3,7,NA,NA,67) what will be the output for the R statement length(X)

Exercise 2

If X = c(NA,3,14,NA,33,17,NA,41) write some R code that will remove all occurrences of NA in X.

a. X[!is.na(X)]

b. X[is.na(X)]

c. X[X==NA]= 0

Exercise 3

If Y = c(1,3,12,NA,33,7,NA,21) what R statement will replace all occurrences of NA with 11?

a. Y[Y==NA]= 11

b. Y[is.na(Y)]= 11

c. Y[Y==11] = NA

Exercise 4

If X = c(34,33,65,37,89,NA,43,NA,11,NA,23,NA) then what will count the number of occurrences of NA in X?

a. sum(X==NA)

b. sum(X == NA, is.na(X))

c. sum(is.na(X))

Exercise 5

Consider the following vector W <- c (11, 3, 5, NA, 6)

Write some R code that will return TRUE for value of W missing in the vector.

Exercise 6

Load 'Orange' dataset from R using the command data(Orange) . Replace all values of age=118 to NA.

Exercise 7

Consider the following vector A <- c (33, 21, 12, NA, 7, 8) .

Write some R code that will calculate the mean of A without the missing value.

Exercise 8

Let:

c1 <- c(1,2,3,NA) ;

c2 <- c(2,4,6,89) ;

c3 <- c(45,NA,66,101) .

If X <- rbind (c1,c2,c3, deparse.level=1) , write a code that will display all rows with missing values.

Exercise 9

Consider the following data obtained from df <- data.frame (Name = c(NA, "Joseph", "Martin", NA, "Andrea"), Sales = c(15, 18, 21, 56, 60), Price = c(34, 52, 21, 44, 20), stringsAsFactors = FALSE)

Write some R code that will return a data frame which removes all rows with NA values in Name column

Exercise 10

Consider the following data obtained from df <- data.frame(Name = c(NA, "Joseph", "Martin", NA, "Andrea"), Sales = c(15, 18, 21, NA, 60), Price = c(34, 52, 33, 44, NA), stringsAsFactors = FALSE)

Write some R code that will remove all rows with NA values and give the following output

Name Sales Price

2 Joseph 18 52

3 Martin 21 33

INDEX VECTORS

Exercise 1

If x <- c("ww", "ee", "ff", "uu", "kk"), what will be the output for x[c(2,3)]?

a. "ee", "ff"

b. "ee"

c. "ff"

Exercise 2

If x <- c("ss", "aa", "ff", "kk", "bb"), what will be the third value in the index vector operation x[c(2, 4, 4)]?

a. "uu"

b. NA

c. "kk"

Exercise 3

If x <- c("pp", "aa", "gg", "kk", "bb"), what will be the fourth value in the index vector operation x[-2]?

a. "aa"

b. "gg"

c. "bb"

Exercise 4

Let a <- c(2, 4, 6, 8) and b <- c(TRUE, FALSE, TRUE, FALSE), what will be the output for the R expression max(a[b])?

Exercise 5

Let a <- c (3, 4, 7, 8) and b <- c(TRUE, TRUE, FALSE, FALSE), what will be the output for the R expression sum(a[b])?

Exercise 6

Write an R expression that will return the sum value of 10 for the vector x <- c(2, 1, 4, 2, 1, NA)

Exercise 7

If x <- c(1, 3, 5, 7, NA) write an r expression that will return the output 1, 3, 5, 7.

Exercise 8

Consider the data frame s <- data.frame(first= as.factor(c("x", "y", "a", "b", "x", "z")), second=c(2, 4, 6, 8, 10, 12)). Write an R statement that will return the

output 2, 4, 10, by using the variable first as an index vector.

Exercise 9

What will be the output for the R expression (c(FALSE, TRUE)) || (c(TRUE, TRUE))?

Exercise 10

Write an R expression that will return the positions of 3 and 7 in the vector x <- c(1, 3, 6, 7, 3, 7, 8, 9, 3, 7, 2).

FACTOR EXERCISES

Exercise 1

If x = c(1, 2, 3, 3, 5, 3, 2, 4, NA), what are the levels of factor(x)?

a. 1, 2, 3, 4, 5

b. NA

c. 1, 2, 3, 4, 5, NA

Exercise 2

Let x <- c(11, 22, 47, 47, 11, 47, 11). If an R expression factor(x, levels=c(11, 22, 47), ordered=TRUE) is executed, what will be the 4th element in the output?

a. 11

b. 22

c. 47

Exercise 3

If z <- c("p", "a" , "g", "t", "b"), then which of the following R expressions will replace the third element in z with "b".

a. factor(z[3]) <- "b"

b. levels(z[3]) <- "b"

c. z[3] <- "b"

Exercise 4

If z <- factor(c("p", "q", "p", "r", "q")) and levels of z are "p", "q" ,"r", write an R expression that will change the level "p" to "w" so that z is equal to: "w", "q" , "w", "r" , "q".

Exercise 5

If:

s1 <- factor(sample(letters, size=5, replace=TRUE)) and s2 <- factor(sample(letters, size=5, replace=TRUE)), write an R expression that will concatenate s1 and s2 in a single factor with 10 elements.

Exercise 6

Consider the iris data set in R. Write an R expression that will 'cut' the Sepal.Length variable and create the following factor with five levels.

(4.3, 5.02] (5.02, 5.74] (5.74, 6.46] (6.46, 7.18] (7.18, 7.9]

32 41 42 24 11

Exercise 7

Consider again the iris data set. Write an R expression that will generate a two-way frequency table with two rows and three colums. The rows should relate to Sepal.length (less than 5: TRUE or FALSE) and columns to Species, with the following output:

setosa versicolor virginica

FALSE 30 49 49

TRUE 20 1 1

Exercise 8

Consider the factor responses <- factor(c("Agree", "Agree", "Strongly Agree", "Disagree", "Agree")), with the following output:

[1] Agree Agree Strongly Agree Disagree Agree

Levels: Agree Disagree Strongly Agree

Later it was found that new a level "Strongly Disagree" exists. Write an R expression that will include "strongly disagree" as new level attribute of the factor and returns the following output:

[1] Agree Agree Strongly Agree Disagree Agree

Levels: Strongly Agree Agree Disagree Strongly Disagree

Exercise 9

Let x <- data.frame(q=c(2, 4, 6), p=c("a", "b", "c")).

Write an R statement that will replace levels a, b, c with labels "fertiliser1", "fertliser2", "fertiliser3".

Exercise 10

If x <- factor(c("high", "low", "medium", "high", "high", "low", "medium")), write an R expression that will provide unique numeric values for various levels of x with the following output:

levels value

1 high 1

2 low 2

3 medium 3

MATRIX EXERCISES

Exercise 1

Create three vectors x,y,z with integers and each vector has 3 elements. Combine the three vectors to become a 3×3 matrix A where each column represents a vector. Change the row names to a,b,c. Think: How about each row represents a vector, can you modify your code to implement it?

Exercise 2

Please check your result from Exercise 1, using is.matrix(A). It should return TRUE, if your answer is correct. Otherwise, please correct your answer. Hint: Note that is.matrix() will return FALSE on a non-matrix type of input. Eg: a vector and so on.

Exercise 3

Create a vector with 12 integers. Convert the vector to a 4*3 matrix B using matrix(). Please change the column names to x, y, z and row names to a, b, c, d. The argument byrow in matrix() is set to be FALSE by default. Please change it to TRUE and print B to see the differences.

Exercise 4

Please obtain the transpose matrix of B named tB .

Exercise 5

Now tB is a 3×4 matrix. By the rule of matrix multiplication in algebra, can we perform tB*tB in R language? (Is a 3×4 matrix multiplied by a 3×4 allowed?) What result would we get?

Exercise 6

As we can see from Exercise 5, we were expecting that tB*tB would not be allowed because it disobeys the algebra rules. But it actually went through the computation in R. However, as we check the output result , we notice the multiplication with a single * operator is performing the componentwise multiplication. It is not the conventional matrix multiplication. How to perform the conventional matrix multiplication in R? Can you compute matrix A multiplies tB ?

Exercise 7

If we convert A to a data.frame type instead of a matrix , can we still compute a conventional matrix multiplication for matrix A multiplies matrix A ? Is there any way we could still perform the matrix multiplication for two data.frame type variables? (Assuming proper dimension)

Exercise 8

Extract a sub-matrix from B named subB . It should be a 3×3 matrix which includes the last three rows of matrix B and their corresponding columns.

Exercise 9

Compute 3*A , A+subB , A-subB . Can we compute A+B? Why?

Exercise 10

Generate a n * n matrix (square matrix) A1 with proper number of random numbers, then generate another n * m matrix A2.

If we have A1*M=A2 (Here * represents the conventional multiplication), please solve for M.

Hint: use the runif() and solve() functions.

E.g., runif(9) should give you 9 random numbers.

BIND EXERCISES

Exercise 1

Try to create matrices from the vectors below, by binding them column-wise. First, without using R, write down whether binding the vectors to a matrix is actually possible; then the resulting matrix and its mode (e.g., character, numeric etc.). Finally check your answer using R.

a. a <- 1:5 ; b <- 1:5

b. a <- 1:5 ; b <- c('1', '2', '3', '4', '5')

c. a <- 1:5 ; b <- 1:4; c <- 1:3

Exercise 2

Repeat exercise 1, binding vectors row-wise instead of column-wise while avoiding any row names.

Exercise 3

Bind the following matrices column-wise. First, without using R, write down whether binding the matrices is actually possible; then the resulting matrix and its mode (e.g., character, numeric etc.). Finally check your answer using R.

a. a <- matrix(1:12, ncol=4); b <- matrix(21:35, ncol=5)

b. a <- matrix(1:12, ncol=4); b <- matrix(21:35, ncol=3)

c. a <- matrix(1:39, ncol=3); b <- matrix(LETTERS, ncol=2)

Exercise 4

Bind the matrix a <- matrix(1:1089, ncol=33) to itself, column-wise, 20 times (i.e., resulting in a new matrix with 21*33 columns). Hint: Avoid using cbind() to obtain an efficient solution. Various solutions are possible. If yours is different from those shown on the solutions page, please post yours on that page as comment, so we can all benefit.

Exercise 5

Try to create new data frames from the data frames below, by binding them column-wise. First, without using R, write down whether binding the data frames is actually possible; then the resulting data frame and the class of each column (e.g., integer, character, factor etc.). Finally check your answer using R.

a. a <- data.frame(v1=1:5, v2=LETTERS[1:5]) ; b <- data.frame(var1=6:10, var2=LETTERS[6:10])

b. a <- data.frame(v1=1:6, v2=LETTERS[1:6]) ; b <- data.frame(var1=6:10, var2=LETTERS[6:10])

Exercise 6

Try to create new data frames from the data frames below, by binding them row-wise. First, without using R, write down whether binding the data frames is actually possible; then the resulting data frame and the class of each column (e.g., integer, character, factor etc.). Finally check your answer using R, and explain any unexpected output.

a. a <- data.frame(v1=1:5, v2=LETTERS[1:5]) ; b <- data.frame(v1=6:10, v2=LETTERS[6:10])

b. a <- data.frame(v1=1:6, v2=LETTERS[1:6]) ; b <- data.frame(v2=6:10, v1=LETTERS[6:10])

Exercise 7

a. Use cbind() to add vector v3 <- 1:5 as a new variable to the data frame created in exercise 6b.

b. Reorder the columns of this data frame, as follows: v1, v3, v2.

Exercise 8

Consider again the matrices of exercise 3b. Use both cbind() and rbind() to bind both matrices column-wise, adding NA for empty cells.

Exercise 9

Consider again the data frames of exercise 5b. Use both cbind() and rbind() to bind both matrices column-wise, adding NA for empty cells.

LIST EXERCISES

Exercise 1

If:

p <- c(2,7,8), q <- c("A", "B", "C") and

x <- list(p, q),

then what is the value of x[2]?

a. NULL

b. "A" "B" "C"

c. "7"

Exercise 2

If:

w <- c(2, 7, 8)

v <- c("A", "B", "C")

x <- list(w, v),

then which R statement will replace "A" in x with "K".

a. x[[2]] <- "K"

b. x[[2]][1] <- "K"

c. x[[1]][2] <- "K"

Exercise 3

If a <- list ("x"=5, "y"=10, "z"=15), which R statement will give the sum of all elements in a?

a. sum(a)

b. sum(list(a))

c. sum(unlist(a))

Exercise 4

If Newlist <- list(a=1:10, b="Good morning", c="Hi"), write an R statement that will add 1 to each element of the first vector in Newlist.

Exercise 5

If b <- list(a=1:10, c="Hello", d="AA"), write an R expression that will give all elements, except the second, of the first vector of b.

Exercise 6

Let x <- list(a=5:10, c="Hello", d="AA"), write an R statement to add a new item z = "NewItem" to the list x.

Exercise 7

Consider y <- list("a", "b", "c"), write an R statement that will assign new names "one", "two" and "three" to the elements of y.

Exercise 8

If x <- list(y=1:10, t="Hello", f="TT", r=5:20), write an R statement that will give the length of vector r of x.

Exercise 9

Let string <- "Grand Opening", write an R statement to split this string into two and return the following output:

[[1]]

[1] "Grand"

[[2]]

[1] "Opening"

Exercise 10

Let:

y <- list("a", "b", "c") and

q <- list("A", "B", "C", "a", "b", "c").

Write an R statement that will return all elements of q that are not in y, with the following result:

[[1]]

[1] "A"

[[2]]

[1] "B"

[[3]]

[1] "C"

DATA FRAME EXERCISES

Exercise 1

Create the following data frame, afterwards invert Sex for all individuals.

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

Exercise 2

Create this data frame (make sure you import the variable Working as character and not factor).

	Working
Alex	Yes
Lilly	No
Mark	No
Oliver	Yes
Martha	Yes
Lucas	No
Caroline	Yes

Add this data frame column-wise to the previous one.

a) How many rows and columns does the new data frame have?

b) What class of data is in each column?

Exercise 3

Check what class of data is the (built-in data set) state.center and convert it to data frame.

Exercise 4

Create a simple data frame from 3 vectors. Order the entire data frame by the first column.

Exercise 5

Create a data frame from a matrix of your choice, change the row names so every row says id_i (where i is the row number) and change the column names to variable_i (where i is the column number). I.e., for column 1 it will say variable_1, and for row 2 will say id_2 and so on.

Exercise 6

For this exercise, we'll use the (built-in) dataset VADeaths.

a) Make sure the object is a data frame, if not change it to a data frame.

b) Create a new variable, named Total, which is the sum of each row.

c) Change the order of the columns so total is the first variable.

Exercise 7

For this exercise we'll use the (built-in) dataset state.x77.

a) Make sure the object is a data frame, if not change it to a data frame.

b) Find out how many states have an income of less than 4300.

c) Find out which is the state with the highest income.

Exercise 8

With the dataset swiss, create a data frame of only the rows 1, 2, 3, 10, 11, 12 and 13, and only the variables Examination, Education and Infant.Mortality.

a) The infant mortality of Sarine is wrong, it should be a NA, change it.

b) Create a row that will be the total sum of the column, name it Total.

c) Create a new variable that will be the proportion of Examination (Examination / Total)

Exercise 9

Create a data frame with the datasets state.abb, state.area, state.division, state.name, state.region. The row names should be the names of the states.

a) Rename the column names so only the first 3 letters after the full stop appear (e.g. States.abb will be abb).

Exercise 10

Add the previous data frame column-wise to state.x77

a) Remove the variable div.

b) Also remove the variables Life Exp, HS Grad, Frost, abb, and are.

c) Add a variable to the data frame which should categorize the level of illiteracy:
[0,1) is low, [1,2) is some, [2, inf) is high.

d) Find out which state from the west, with low illiteracy, has the highest income, and what that income is.

READING DELIMITED DATA

For each exercise we provide a data set that can be accessed through the link shown in the exercise. You can read the data from this link directly (clicking on it will show the url in the address bar of your browser),

or you can download the data first to a local directory, and read it from there.

Exercise 1

Read the file Table0.txt.

a) Change the names of the columns
to Name, Age, Height, Weightand Sex.

b) Change the row names so that they are the same as Name, and remove the variable Name.

Exercise 2

Read the file Table1.txt, how many rows and columns does it have?

a) Reread the file and make the variable Name be the row names. Make sure you read the variable as characters and not as factors.

Exercise 3

Read the file Table2.txt, watch out for the first line.

Exercise 4

Read the file Table3.txt, watch out for the first line and the missing values.

Exercise 5

Read the file Table4.txt, watch out for the missing values and the decimal separator.

Exercise 6

Read the file Table5.txt, watch out for the missing values and the decimal separator and the separator.

Exercise 7

Read the file states1.csv, the names of the states should be the row names.

Exercise 8

Read the file states2.csv, the names of the states should be the row names, watch out for the decimal separator and the separator.

Exercise 9

Read the file states2.csv, watch out for the same as the last exercise plus the missing values. Add to the previous dataset, column-wise.

Exercise 10

Read the file Table6.txt, check out the file first. Notice that the information is repeated, we only want the first non-repeated ones. Make sure to create only characters not factors this time around. Lastly, we don't want the comments.