Universitatea Tehnică a Moldovei

Facultatea Calculatoare Informatică şi Microelectronică

Departamentul Ingineria Software și Automatică

**RAPORT**

Lucrarea de laborator nr. 2

# La disciplina „Programarea Declarativa”

Tema: “Vectori, matrici si data frame-uri.”

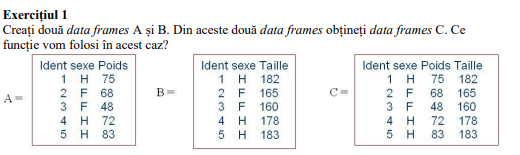
A efectuat: st. gr. SI-211 S. Chirita

A verificat: V. Rusu

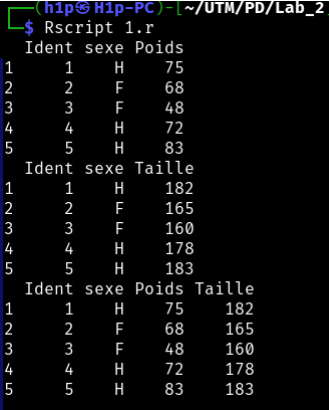
Chișinău – 2023

Lucrarea de laborator nr. 2

#### Sarcina de lucru



|  |
| --- |
| A<-data.frame(Ident=c(1,2,3,4,5), sexe=c("H","F","F","H","H"), Poids=c(75,68,48,72,83))  B<-data.frame(Ident=c(1,2,3,4,5), sexe=c("H","F","F","H","H"), Taille=c(182,165,160,178,183))  print(A)  print(B)  C<-merge(A,B, by = c("Ident","sexe"), all=TRUE)  print(C) |



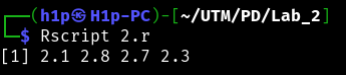
Funcția merge() în R vă permite să specificați opțiuni precum tipul de îmbinare (inner, outer, left, right) și coloanele pe care doriți să le folosiți pentru îmbinare, în funcție de nevoile dvs.

**Exercițiul 2**

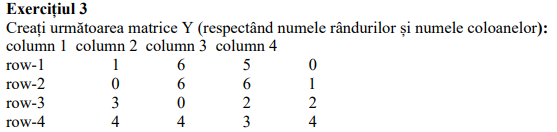
Extrageți toate numerele între 2 și 3 din următorul vector:

> x=c(0.2, 0.6, 2.1, 3.7, 2.8, 2.7, 1.9, 2.3, 5.9)

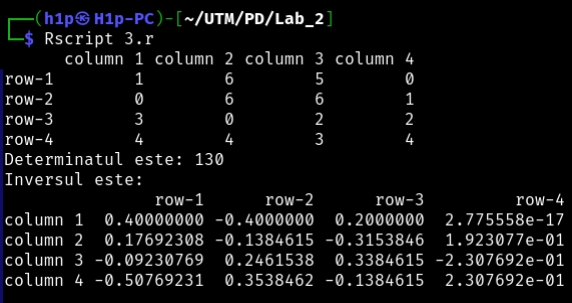
|  |
| --- |
| x=c(0.2, 0.6, 2.1, 3.7, 2.8, 2.7, 1.9, 2.3, 5.9)  result<-x[x>=2&x<=3]  print(result) |



Sarcina este de a identifica și extrage toate numerele din acest vector care se încadrează între valorile 2 și 3. În acest caz, numerele care satisfac condiția dată sunt 2.1, 2.8, 2.7 și 2.3. Deci, rezultatul extragerii va consta într-un alt vector care conține aceste numere.



|  |
| --- |
| Y <- matrix(c(1, 0, 3, 4, 6, 6, 0, 4, 5, 6, 2, 3, 0, 1, 2, 4), nrow = 4, dimnames = list(c("row1", "row-2", "row-3", "row-4"), c("column 1", "column 2", "co lumn 3", "column 4")))  print(Y)  determitatY<-det(Y)  cat("Determinatul este:",determitatY,"\n")  inverseY <- solve(Y)  cat("Inversul este:","\n")  print(inverseY) |



Pentru a realiza aceste sarcini, putem utiliza funcțiile det() pentru determinant și solve() pentru invers în limbajul R.

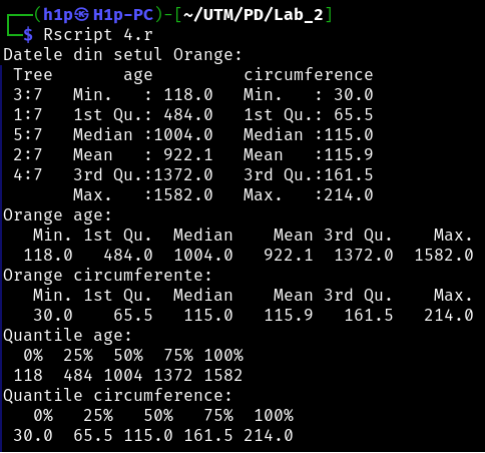
**Exercițiul 4**

a) Încărcați datele „Orange” (disponibile în R). Calculați statisticile de bază (media, abaterea standard, min, etc.) ale ultimelor două variabile ale acestui set de date.

b) Calculați quartilele ambelor variabile.

c) Folosind funcția “apply”, calculați toate decilele ambelor variabile folosind argumentul „probs” al funcției „quantile”.

|  |
| --- |
| cat("Datele din setul Orange:\n")  summary(Orange)  cat("Orange age:\n")  summary(Orangeage)  cat(′′Orangecircumferente : \n′′)  summary(Orangecircumference)  Q\_age<-quantile(Orangeage)  Qcircumference < −quantile(Orangecircumference)  cat("Quantile age:\n") print(Q\_age)  cat("Quantile circumference:\n")  print(Q\_circumference) |



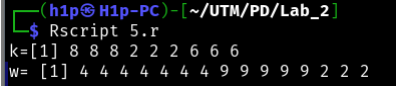
Acest exercițiu implică manipularea și analiza datelor din setul de date "Orange" disponibil în R.

**Exercițiul 5**

a) Creați vectorul k format din de trei ori șirul de numere (8; 2; 6).

b) Creați vectorul w compus din de șapte ori numărul 4, de 5 ori numărul 9 și de 3 ori numărul 2 (prin două metode diferite).

|  |
| --- |
| k<-c(8,2,6)  k<-rep(k, each=3)  cat("k=")  print(k)  w<-c(rep(4,7), rep(9,5), rep(2,3))  cat("w=") print(w) |



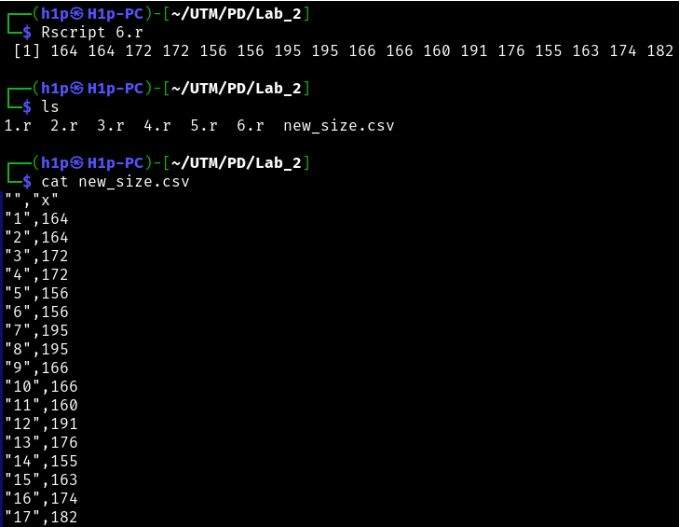
**Exercițiul 6**

a) Introduceți variabila „ size” care conține următoarele 9 valori: 178, 175, 160, 191, 176, 155, 163, 174, 182.

b) Introduceți variabila “size\_1” care conține următoarele 5 valori : 164, 172, 156, 195, 166. c) Din variabilele „size” și „size\_1”, creați variabila „new.size” care conține: cele cinci valori ale „size1” repetate de două ori și ultimele șapte valori ale „size” .

d) Salvați în directorul dvs. de lucru variabila „new.size” într-un fișier în format .csv.

|  |
| --- |
| size<-c(178,175,160,191,176,155,163,174,182)  size\_1<-c(164, 172, 156, 195, 166)  new.size <- c(rep(size\_1, each = 2), tail(size, 7))  print(new.size)  write.csv(new.size, file = "new\_size.csv") |

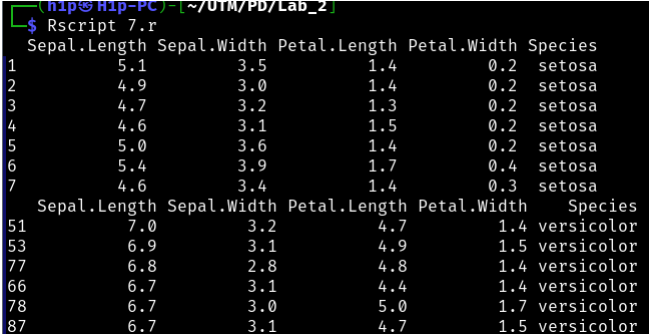


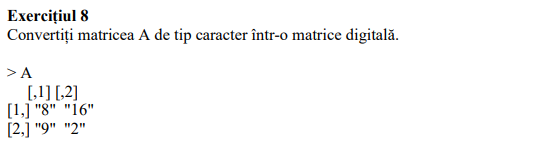
**Exercițiul 7**

a) Încărcați setul de date „iris”, apoi vizualizați primele 7 linii. Creați un subset de date care să conțină doar datele din modalitatea „versicolor” a variabilei „ Species ” (numiți acest nou set de date „new.iris”).

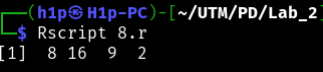
b) Sortați în ordine descrescătoare datele „new.iris” în funcție de variabila Sepal.Length.

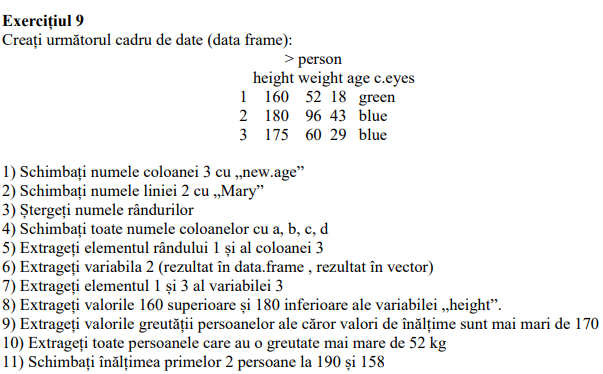
|  |
| --- |
| data("iris")  head(iris,7)  new.iris <- subset(iris, Species == "versicolor")  new.iris <- new.iris[order(-new.iris$Sepal.Length), ]  head(new.iris) |



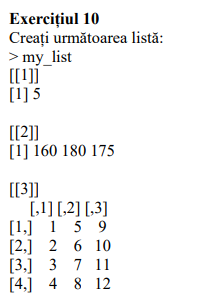


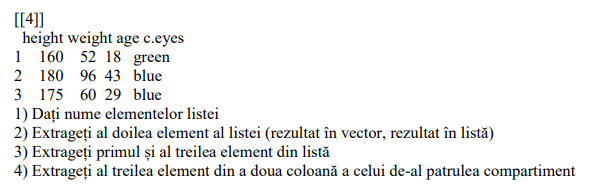
|  |
| --- |
| A <- matrix(c("8", "16", "9", "2"), nrow = 2, ncol = 2)  A\_digital <- as.numeric(A)  print(A\_digital) |



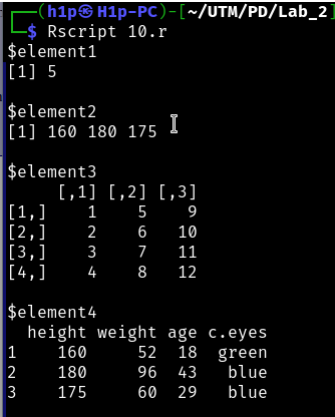


|  |
| --- |
| person <- data.frame( height = c(160, 180, 175), weight = c(52, 96, 60), age = c(18, 43, 29), c.eyes = c("green", "blue", "blue") )  colnames(person)[3] <- "new.age"  rownames(person)[2] <- "Mary"  rownames(person) <- NULL  colnames(person) <- c("a", "b", "c", "d")  element\_1\_3 <- person[1, 3]  variabila\_2\_df <- person[, "b"]  variabila\_2\_vector <- person"b"  elemente\_1\_si\_3\_age <- person[c(1, 3), "c"]  height\_160\_superioare <- personheight[personheight > 160]  height\_180\_inferioare <- personheight[personheight < 180]  greutate\_inalta <- personweight[personheight > 170]  persoane\_greutate\_mai\_mare <- person[person$weight > 52, ]  person[1:2, "a"] <- c(190, 158) |





|  |
| --- |
| my\_list <- list( 5, c(160, 180, 175), matrix(1:12, ncol = 3), data.frame( height = c(160, 180, 175), weight = c(52, 96, 60), age = c(18, 43, 29), c.eyes = c("green", "blue", "blue") ) )  names(my\_list) <- c("element1", "element2", "element3", "element4")  element\_2\_vector <- my\_list[2]  element\_2\_lista <- list(my\_list[2])  elemente\_1\_si\_3 <- my\_list[c("element1", "element3")]  element\_3\_al\_doua\_coloana <- my\_list4[3, 2]  print(my\_list) |



**Concluzie:** În încheiere, tema vectorilor, matricilor și data frame-urilor este fundamentală în analiza și manipularea datelor. Aceste concepte oferă structuri și instrumente esențiale pentru lucrul cu datele într-un format tabular sau multidimensional, ceea ce facilitează prelucrarea, analiza și vizualizarea acestora. Cunoașterea adecvată a acestor concepte este crucială în domeniul analizei datelor și științei datelor pentru a extrage înțelesuri din date și a lua decizii informate.