**Introdução à Análise de Dados – G 1 (Modelo 1) – 2023.1**

**Nome:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Matrícula:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ IP:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Você recebeu o enunciado da G1 que deve ser **resolvida em sala de aula e enviada pela plataforma EAD.**

A prova contém quatro questões e uma questão bônus sem ônus. Ela é individual, tem a duração de duas horas e todas as atividades relacionadas à solução do trabalho proposto devem ser realizadas, respeitando-se o código de ética do CTC disponível na plataforma EAD.

**Para cada questão da prova deverá ser criado um script correspondente que deverá ser salvo com o nome “INF1514\_G1\_QX\_MATRICULA.R”, substituindo o texto “MATRICULA” pelo número da sua matrícula e X pelo número da questão. Cada script deverá conter todas as implementações realizadas para a correspondente questão, incluindo os testes, sendo que a criação e organização dos scripts faz parte da resolução do trabalho. Nas questões que possuem subitens (a), b), c), ...) identifique cada subitem através de comentários no código.**

1) Para calcular o montante final gerado por seus investimentos um analista utiliza a seguinte fórmula: (2,5 pontos)

Elabore uma função chamada **calculaMontanteFinal** que calcula o montante final de um investimento tendo como parâmetros: o **capital investido**; a **taxa de jutos** anual; e o **tempo de investimento** em anos. A seguir, no mesmo script, siga os seguintes passos:

1. Crie um vetor chamado **ano** que contenha uma sequência de valores de **1** a **10**, com intervalo de **1**, representando possíveis **anos** de **tempo de investimento**;
2. Faça uso de um dos comandos do R que permitem criar ciclos para chamar a função **calculaMontanteFinal** utilizando como parâmetros **capital investido** de **R$ 1000,00**, **taxa de juros** anual de **3%** (**0.03**) e **tempo de investimento** representado pelos elementos do vetor **ano**. Apresente na tela média dos valores retornados pela função **calculaMontanteFinal** para as chamadas realizadas.

calculaMontanteFinal <- function (capital, taxa, tempo)

{

retorno <- 0.0

retorno <- capital \* ((1 + taxa)^tempo)

return(retorno)

}

capitalinicial <- 1000

taxajuros <- 0.03

ano <- 1:10

montante <- rep(0, length(ano))

indice <- 0

for (i in ano)

{

indice <- indice + 1

montante[indice] <- calculaMontanteFinal(capitalinicial, taxajuros, ano[indice])

}

print(mean(montante))

2) Considere um vetor **v** formado pelos números pares mariores ou iguais a **2** e menores ou iguais a **40**. (2,5 pontos)

a) Crie o vetor **v**.

v <- seq(2, 40, 2)

v

b) Calcule a soma dos elementos do vetor **v**.

soma <- sum(v)

soma

c) Selecione os elementos deste vetor que são múltiplos de **5** e os armazene em uma variável chamada **v.selecionado**.

v.selecionado <- v[v%%5 == 0]

v.selecionado

d) Apresente um vetor chamado **v.dividido** que contém a divisão por elemento da sequência obtida no item **a)** por **4**.

v.dividido <- v / 4

v.dividido

e) Calcule a média dos elementos do vetor gerado a partir de **v + v.dividido**.

media <- mean(v + v.dividido)

media

3) Usando as instruções **for** e **if** e as funções **nrow()** e **ncol()**, construa uma função chamada **calcula.soma** que recebe uma matriz **m** como parâmetro e calcula a soma dos elementos da matriz que possuem valor superior a **8**. Crie uma matriz para testar a função **calcula.soma** e apresente na tela o valor retornado pela função. (2,0 pontos)

calcula.soma <- function (m)

{

soma <- 0

for (i in 1:nrow(m)) {

for (j in 1:ncol(m)) {

if (m [i, j] > 8) {

soma <- soma + m [i, j]

}

}

}

return (soma)

}

m <- matrix (1:12, ncol = 4)

print(calcula.soma (m))

4) Uma loja que vende carros usados está avaliando os preços e quantidades vendidas de um determinado modelo. Os valores estão na tabela a seguir: (3,0 pontos)

|  |  |  |
| --- | --- | --- |
| **Ano** | **Preço de venda (R$)** | **Quantidade vendida** |
| 2019 | 26.200,00 | 21 |
| 2020 | 27.800,00 | 25 |
| 2021 | 28.200,00 | 22 |
| 2022 | 31.500,00 | 26 |

Para os dados acima e usando os recursos da linguagem R:

a) Crie um data frame chamado **carro** contendo três colunas a saber: **Ano**, com os anos avaliados; **PrecoVenda**, com o preço médio de venda por ano; **Quantidade**, com a quantidade vendida por ano.

Ano <- c (2019, 2020,2021, 2022)

PrecoVenda <- c (26200, 27800, 28200, 31500)

Quantidade <- c (21, 25, 22, 26)

carro <- data.frame (Ano, PrecoVenda, Quantidade)

carro

b) Usando funções do R, ordene o data frame pela coluna **Quantidade**.

carro <- carro [ order (carro$Quantidade), ]

carro

c) Usando funções do R, crie no data frame uma coluna chamada **Receita**, com a receita das vendas por ano. A receita das vendas por ano é obtida multiplicando-se o preço de venda pela quantidade vendida.

carro$Receita <- (carro$PrecoVenda \* carro$Quantidade)

carro

d) Usando funções do R, calcule a quantidade média de carros vendidos, ao longo dos anos, usando o data frame criado no item a).

media <- mean (carro$Quantidade)

media

e) Selecione no data frame criado no item a) apenas as colunas **Ano** e **Quantidade** e as linhas cujo preço de venda seja superior a **R$ 26.600,00**. Devem ser utilizados mecanismos ou funções de filtro do R.

carro.f <- carro [carro $PrecoVenda > 26600, ]

carro.f <- carro.f [, c("Ano","Quantidade")]

carro.f

5) (**Bônus sem ônus**) Elabore uma função chamada **encontra.valor** que recebe dois vetores numéricos **x** e **y** e retorna o total de elementos do primeiro vetor que também estão no segundo vetor. Considere também que: se um mesmo elemento de **x** aparecer mais de uma vez em **y**, deverá ser contado como uma única vez; se **x** apresentar elementos repetidos, eles deverão ser considerados como elementos distintos. Teste a função criada. (1,0 ponto)

encontra.valor = function (x, y)

{

total <- 0

i <- 0

while (i < length(x)) {

i <- i + 1

j <- 0

while (j < length(y)) {

j <- j + 1

if (x [i] == y[j]) {

total <- total + 1

break

}

}

}

print (total)

}

x <- c (1, 5, 1, 100)

y <- c (0, 1, 2, 5, 3, 4, 5, 6, 7, 8, 9, 5)

encontra.valor (x, y)