

```
# Lista de Exercícios 1
# Aluno: Carlos Eduardo Fontaneli
# RA: 769949
# Curso: Bacharelado em Ciência da Computação
```

```
library(tidyverse)
library(stringr)
```

```
# Questão 1
distancia_euclidiana <- function(v1, v2, v3, v4) {
  sqrt((v3 - v1)^2 + (v4 - v2)^2)
}
```

```
# Questão 2
converte <- function(valor) {
  if (valor %% 1 != 0) {
    print("Erro parâmetro não é inteiro.")
  } else {
    cem <- valor %% 100
    valor <- valor - (100 * cem)

    cinquenta <- valor %% 50
    valor <- valor - (50 * cinquenta)

    dez <- valor %% 10
    valor <- valor - (10 * dez)

    cinco <- valor %% 5
    valor <- valor - (5 * cinco)

    dois <- valor %% 2
    valor <- valor - (2 * dois)

    um <- valor

    lista_final <- list(
      notas_cem = cem,
      notas_cinquenta = cinquenta,
      notas_dez = dez,
      notas_cinco = cinco,
      notas_dois = dois,
      notas_um = um
    )
  }
}
```

```
# Questão 3
checa_coluna_linha <- function(index, coluna, tabuleiro) {
  if (coluna) {
    aux <- tabuleiro[, index]
  } else {
```

```

    aux <- c(tabuleiro[index, ])
  }

  if (any(duplicated(aux) == 1)) {
    return(0)
  } else if (any(aux < 0 | aux > 9) == 1) {
    return(0)
  } else {
    return(1)
  }
}

checa_quadrado_menor <- function(tabuleiro) {
  for (coluna in range(0, 9, 3)) {
    for (linha in range(0, 9, 3)) {
      aux <- c()
      for (i in range(0, linha + 3)) {
        for (j in range(0, coluna + 3)) {
          aux <- c(aux, tabuleiro[i, j])
        }
      }
      if (any(duplicated(aux) == 1)) {
        return(0)
      }
    }
  }
  return(1)
}

checa_tabuleiro <- function(tabuleiro) {
  if (nrow(tabuleiro) + ncol(tabuleiro) != 18) {
    return(FALSE)
  }
  for (i in 1:9) {
    validacao1 <- checa_coluna_linha(i, 1, tabuleiro)
    validacao2 <- checa_coluna_linha(i, 0, tabuleiro)
    if (validacao1 + validacao2 != 2) {
      return(FALSE)
    }
  }
  validacao3 <- checa_quadrado_menor(tabuleiro)
  if (validacao3 == 0) {
    return(FALSE)
  } else {
    return(TRUE)
  }
}

# Questao 4
completa_menos_um <- function(tabuleiro) {
  for (coluna in 1:9) {
    for (linha in 1:9) {
      if (tabuleiro[linha, coluna] == -1) {

```

```

        tabuleiro[linha, coluna] <- 1
        while (!checa_tabuleiro(tabuleiro)) {
            tabuleiro[linha, coluna] <- tabuleiro[linha, coluna] + 1
        }
    }
}
return(tabuleiro)
}

```

Questao 5

```
dados <- read_csv("/home/fonta42/Desktop/ICDuR/Listas/imdb_top_1000.csv")
```

a)

```
filmes_drama <- filter(dados, str_detect(Genre, "Drama"))
filmes_drama
```

b)

```
filmes_freeman_ordenado <- arrange(
  filter(
    dados,
    Star1 == "Morgan Freeman" |
    Star2 == "Morgan Freeman" |
    Star3 == "Morgan Freeman" |
    Star4 == "Morgan Freeman"
  ),
  Released_Year
)
filmes_freeman_ordenado
```

c)

```
imdb_nolan <- select(
  filter(
    dados,
    Director == "Christopher Nolan"
  ),
  IMDB_Rating
)
media_imdb_nolan <- sapply(imdb_nolan, mean, na.rm = TRUE)
media_imdb_nolan
```

d)

```
cinco_melhores_pitt <- head(
  arrange(
    filter(
      dados,
      Star1 == "Brad Pitt" |
      Star2 == "Brad Pitt" |
      Star3 == "Brad Pitt" |
      Star4 == "Brad Pitt"
    )
  )
)
```

```

    ),
    desc(IMDB_Rating)
  ),
  n = 5
)
cinco_melhores_pitt

# e)
ano_lancamento_acao <- select(
  filter(
    dados,
    str_detect(Genre, "Action")
  ),
  Released_Year
)
ano_lancamento_acao <- as.numeric(unlist(ano_lancamento_acao))

num_votos_acao <- select(
  filter(
    dados,
    str_detect(Genre, "Action")
  ),
  No_of_Votes
)
media_ano_lancamento_acao <- mean(ano_lancamento_acao)
media_num_votos_acao <- sapply(num_votos_acao, mean, na.rm = TRUE)
media_num_votos_acao

# f)
filmes_drama <- filter(dados, str_detect(Genre, "Drama"))
filmes_drama$Type <- "Drama"

filmes_comedia <- filter(dados, str_detect(Genre, "Comedy"))
filmes_comedia$Type <- "Comedy"

filmes_drama_comedia <- filter(dados, str_detect(Genre, "Comedy, Drama"))
filmes_drama_comedia$Type <- "Comedy and Drama"

filmes <- rbind(filmes_drama, filmes_comedia, filmes_drama_comedia)
filmes <- group_by(filmes, Type, Released_Year)
filmes <- summarise(filmes, "Contagem" = count(filmes, Released_Year))
filmes

grafico_filmes_linha_ponto <- ggplot(data = filmes) +
  geom_line(mapping = aes(
    x = Contagem$Released_Year,
    y = Contagem$n,
    group = Contagem$Type,
    colour = Contagem$Type,
  )) +
  scale_x_discrete(breaks = seq(1920, 2020, by = 10)) +
  labs(x = "Ano", y = "Filmes Lançados") +
  theme(

```

```

    axis.title = element_text(size = 10),
    plot.title = element_text(
      size = 12,
      face = "bold"
    )
  ) +
  ggtitle("Lançamento de filmes de comédia e/ou drama por ano")
grafico_filmes_linha_ponto

```

```

# g)
star_wars_filmes <- filter(dados, str_detect(Series_Title, "Star Wars"))
star_wars_filmes$Title <- str_wrap(star_wars_filmes$Series_Title, width =
25)

grafico_star_wars_filmes <- ggplot(data = star_wars_filmes) +
  geom_col(
    stat = "identity",
    position = position_dodge(),
    mapping = aes(
      x = Title,
      y = No_of_Votes,
      fill = Series_Title
    )
  ) +
  scale_y_continuous(n.breaks = 10) +
  labs(x = "Filme", y = "Votos Recebidos") +
  theme(
    axis.title = element_text(size = 10),
    plot.title = element_text(size = 12, face = "bold")
  ) +
  ggtitle("Votos por filme da franquia Star Wars")
grafico_star_wars_filmes

```