

Homework Assignment (1)

Task A

```
# Set seed value for reproducibility
set.seed(10)
range <- -10:30
size <- 20

# Create four vectors with 20 random integer values each (between -10 and 30)
v1 <- sample(range, size = size)
v2 <- sample(range, size = size)
v3 <- sample(range, size = size)
v4 <- sample(range, size = size)

# Combine the vectors above to create a matrix (column-wise)
mat <- matrix(c(v1, v2, v3, v4), nrow = 20, ncol = 4)

# Print the matrix
print(mat)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  -2  18  17  22
## [2,]  -1  30  16  24
## [3,]   1  10   3  -1
## [4,]  -3  17   2  28
## [5,]   8  15  22   5
## [6,]  13  22  14   4
## [7,]   4  -4  -3  20
## [8,]  24  26   5  27
## [9,]  -4  19  15  14
## [10,] 29  14  25   7
## [11,] -9  -7   6  -8
## [12,] 25  29  23  11
## [13,]  2   7  12  23
## [14,] 27  13   4  18
## [15,]  3   8  28  16
## [16,] 22  11  13  30
## [17,] -5  25  -9  25
## [18,] 15   0  -7  10
## [19,] 11   4  -8   2
## [20,]  7 -10  29  17
```

```
# Print the size of the matrix
print(dim(mat))
```

```
## [1] 20  4
```

```
# Print the transpose of the matrix
print(t(mat))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14]
## [1,]   -2  -1    1   -3    8   13    4   24   -4   29   -9   25    2   27
## [2,]   18   30   10   17   15   22   -4   26   19   14   -7   29    7   13
## [3,]   17   16    3    2   22   14   -3    5   15   25    6   23   12    4
## [4,]   22   24   -1   28    5    4   20   27   14    7   -8   11   23   18
##      [,15] [,16] [,17] [,18] [,19] [,20]
## [1,]      3    22   -5    15    11     7
## [2,]      8    11   25     0     4   -10
## [3,]     28    13   -9    -7    -8    29
## [4,]     16    30   25    10     2    17
```

Task B

```
# Create a function to find the common elements in two vectors
find_common_elements <- function(x, y) {
  return (intersect(x, y))
}
```

```
# Sample input 1 (integers)
x <- 1:10
y <- 8:11
common <- find_common_elements(x, y)
```

```
print(common)
```

```
## [1]  8  9 10
```

```
# Sample input 2 (strings)
x <- c("fort wayne", "chicago", "indianapolis", "austin")
y <- c("chicago", "fort wayne", "dallas", "boston", "cleveland")
common <- find_common_elements(x, y)
```

```
print(common)
```

```
## [1] "fort wayne" "chicago"
```

```
# Sample input 3 (boolean)
x <- c(T, T, F, F, F, T, T)
y <- c(F, F, F, F, T, T, T)
common <- find_common_elements(x, y)
```

```
print(common)
```

```
## [1] TRUE FALSE
```

Task C

```
# i. Create the data frame
column_names <- c("name", "type", "diameter", "rotation", "rings")
c1 <- c("Mercury", "Venus", "Earth", "Mars", "Jupiter", "Saturn", "Uranus", "Neptune")
```

```
c2 <- c("Terrestrial planet", "Terrestrial planet", "Terrestrial planet", "Terrestrial planet", "Gas gi
```

```
c3 <- c(0.382, 0.949, 1.000, 0.532, 11.209, 9.449, 4.007, 3.883)
```

```
c4 <- c(58.64, -243.02, 1.00, 1.03, 0.41, 0.43, -0.72, 0.67)
```

```
c5 <- c(F, F, F, F, T, T, T, T)
```

```
planet_df <- data.frame(c1, c2, c3, c4, c5)
names(planet_df) <- column_names
attach(planet_df)
```

```
# Print the data frame
print(planet_df)
```

```
##      name      type diameter rotation rings
## 1 Mercury Terrestrial planet    0.382    58.64 FALSE
## 2  Venus Terrestrial planet    0.949   -243.02 FALSE
## 3  Earth Terrestrial planet    1.000     1.00 FALSE
## 4   Mars Terrestrial planet    0.532     1.03 FALSE
## 5 Jupiter      Gas giant   11.209     0.41  TRUE
## 6  Saturn      Gas giant    9.449     0.43  TRUE
## 7  Uranus      Gas giant    4.007    -0.72  TRUE
## 8 Neptune      Gas giant    3.883     0.67  TRUE
```

```
# ii. Print the diameter of Venus
print(planet_df$diameter[name == "Venus"])
```

```
## [1] 0.949
```

```
# iii. Print the details for the planet with the largest diameter
print(planet_df[diameter == max(diameter), ])
```

```
##      name      type diameter rotation rings
## 5 Jupiter Gas giant   11.209     0.41  TRUE
```

```
# iv. Print the names of gas giants
print(planet_df$name[type == "Gas giant"])
```

```
## [1] "Jupiter" "Saturn"  "Uranus"  "Neptune"
```

```
# v. Sort the data frame by diameter
print(planet_df[order(diameter, decreasing = T), ])
```

##	name	type	diameter	rotation	rings
## 5	Jupiter	Gas giant	11.209	0.41	TRUE
## 6	Saturn	Gas giant	9.449	0.43	TRUE
## 7	Uranus	Gas giant	4.007	-0.72	TRUE
## 8	Neptune	Gas giant	3.883	0.67	TRUE
## 3	Earth	Terrestrial planet	1.000	1.00	FALSE
## 2	Venus	Terrestrial planet	0.949	-243.02	FALSE
## 4	Mars	Terrestrial planet	0.532	1.03	FALSE
## 1	Mercury	Terrestrial planet	0.382	58.64	FALSE